Evaluating Entity Relationship Recommenders in a Complex Information Retrieval Context

by

Jack Thomas

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Masters of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2014

AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Information Retrieval, as a field, has long subscribed to an orthodox evaluation approach known as the Cranfield paradigm. This approach and the assumptions that underpin it have been essential to building the traditional search engine infrastructure that drives today's modern information economy. In order to build the information economy of tomorrow, however, we must be prepared to reexamine these assumptions and create new, more sophisticated standards of evaluation to match the more complex information retrieval systems on the horizon.

In this thesis, we begin this introspective process and launch our own evaluation method for one of these complex IR systems, entity-relationship recommenders. We will begin building a new user model adapted to the needs of a different user experience. To support these endeavors, we will also conduct a study with a mockup of our complex system to collect real behavior data and evaluation results. By the end of this work, we shall present a new evaluative approach for one kind of entity-relationship system and point the way for other advanced systems to come.

Acknowledgements

The author wishes to thank his supervisor, Olga Vechtomova, for her guidance. Without her, this thesis would not have been possible. Acknowledgements should also be made toward the other members of the thesis committee for their time. The participants who took part in the user study critical to the research presented herein are also richly deserving of thanks

There are others beyond the academic no less deserving of praise. The author would also like to thank his parents and sister, as well as his whole extended family, for their regular support. Friends in the PLG lab, in the rest of the university and back home in Fredericton have helped to make the terms of this Master's degree some of the best a student could ask for. All have played a critical role in bringing the author to this point.

We stand on the shoulders of giants.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

At every moment of every day, someone is struck with a new need for information. Perhaps they can't remember who starred in a certain film, or they need the location of the local hockey rink – something innocuous, direct. Within moments, our curious individual can take out a phone or open a laptop, type up their question and get an immediate, accurate response. This kind of unprecedented access to information is the cornerstone of modern society.

Just as commonly, however, someone is struck with a more nuanced need for information. Maybe they need to familiarize themselves with a period of history, or have taken interest in a new hobby and want to get their bearings. These are not needs that can be answered with a single query but rather a wider exploration, something not yet well-supported by our existing information infrastructure. The onus rests with the user to coordinate their queries, infer relationships and identify entities of note.

A new generation of information retrieval systems is poised to meet this higher-level need. One of the paradigms being brought to bear on the problem is the entity-relationship model, where entities (people, places, events, etc.) are imagined as nodes on a graph while relationships between them are the connecting edges. While the model has been around in one form or another for decades, new implementations are being considered to meet these needs.

Taking on the challenge of higher-order search tasks is not as simple as dreaming up new tools, however. For all these new systems, there must be some means to measure their progress. Complex IR tasks by definition exceed the boundaries of the current evaluation paradigm, knitting together multiple queries and information needs into a single activity. To advance this cause, we must put forward a new theory of evaluation, supporting new methods and metrics.

Even within the entity relationship model, there are a number of different complex IR systems that suggest themselves, each with their own evaluative needs. Our goal is to produce a method to evaluate entity relationship recommender systems, an implementation where the user interacts with an input entity for the topic they are interested in and the system recommends relationships between that entity and others based on how relevant they are to the user's interests. Clicking these relationships

will add the entities they connect with to the user's own subgraph, opening up new relationships which connect to even more entities.

The goal is to encourage exploration by presenting users with interesting relationships between the entities that make up their subject of interest. As users build their own subgraph, they get a strong visualization of how different elements of their topic hang together. Evaluating the success of these recommender systems presents a challenge, as it must account for the relevance of recommended relationships to a user's interests and how well they support the user in navigating the graph.

Methods cannot spring out of speculation fully formed. That is why this thesis will also include a study where participants interact with a mocked up entity relationship recommender system. By allowing them to investigate academic subjects using this facsimile system, we gain valuable data and insight regarding how users explore entity relationship graphs. The study's purpose is twofold, as their relevance judgments at the end of the study also provide the means by which we will test the performance of our method in an effort to correlate it with real-world outcomes.

The contributions of this thesis are divided among our two subjects, behavior and evaluation. On the behavior side, we investigate current models used in information retrieval to determine if their assumptions and abstractions can be adapted to these new, more advanced IR applications. We then take our study findings and begin roughing out a new behavior model, one tuned more to the goals of exploratory search and the different behavior exhibited by users when interacting with complex systems.

On evaluation, we tackle the specific problem of developing a new method of evaluation for entity-relationship recommender systems. This involves laying out the requirements and components of the system, identifying the exploratory behavior it is meant to support and developing a set of clear, objective metrics to measure how well it achieves this task. It will also involve applying what we have learned about behavior in the previous chapter, as well as validating our method with data from our study.

It is important at this time to note that we cannot hope to solve the entire question of evaluation for information retrieval in this one thesis. Our aim is merely to advance the subject into new areas and grant new systems a means for their assessment. The specific contributions of this thesis can be listed as follows:

1. Survey the history of information retrieval evaluation. Examine both existing orthodoxy and alternatives. Explain entity relationship recommender systems.

2. Conduct a study on entity relationship recommender systems. Collect and analyze data concerning user's behavior and their assessments of relevance.

3. Begin developing a user model for users engaging with entity relationship systems. Test traditional IR assumptions. Seek new insights from user data.

4. Develop one method for a specific breed of system, the entity relationship recommender. Provide evidence of its merit. Determine if this may point the way for future complex IR evaluation methods.

The rest of this thesis is organized into five chapters. Chapter two will cover the history of evaluation in information retrieval, provide context about the specific entity-relationship system we seek to evaluate and outline related works in alternative IR evaluation. Chapter three will cover the conduct of the study, explaining why and to what end it is undertaken as well as how it was implemented. The results will be analyzed in the next two chapters, where chapter four investigates the implications for user behavior while chapter five builds on this in order to build and test our evaluation method. Lastly, chapter six will organize our conclusions and present a crisp recounting of the thesis's contributions.

Chapter 2
Related Work

Evaluation in information retrieval has a rich history, with a number of important researchers and advances that have shaped the modern understanding of what a successful information retrieval system is. It would be impossible to present a new alternative without firmly grasping the current standard, which is why this chapter will take the time to introduce traditional IR evaluation. We will follow its history, examine its metrics and measures and examine the underpinning evaluation philosophy at work.

While looking at the current standard is a good start, if we are to build an alternative method, it might be useful to look at other alternatives that have emerged over the years. By considering what forces drove their creation, what evaluation questions they answer and what means they used to distinguish themselves from evaluation orthodoxy we might recognize a path for our own evaluation.

Once we have an appreciation for what it takes to build alternative evaluation methods, we can start turning those principles toward the evaluation of entity relationship systems. Of course, this will first require a greater understanding of how entity relationship systems work and what attempts have been made to evaluate them so far, topics that will be covered by the third section of this chapter.

2.1 Traditional Information Retrieval Evaluation

Information retrieval is a field of great import to the world today. Businesses, governments, universities and the public at large depend on tools like search engines to provide the most relevant and timely returns possible. With such powerful vested interests at work, information retrieval has produced a strong body of work dedicated to evaluation.

2.1.1 The Cranfield Paradigm

Where modern information retrieval began is a debatable point, but one significant contender is put forward in *The Philosophy of Information Retrieval Evaluation* by Voorhees (Voorhees, 2001). She points us to the year 1957 and Cranfield University. There, Cyril Cleverdon conducted what would come to be known as the Cranfield experiments and eventually lead to what is named the Cranfield

paradigm. The original experiments intended nothing so grand, but were instead merely an evaluation of the efficiency of indexing systems at the University's library.

Cleverdon laid out a simple, intuitive yet powerful test regime meant to determine how quickly and effectively different indexing systems could return relevant documents for different queries. To that end, he created a test collection of example queries and determined which documents from the overall collection were relevant to which queries. Relevance was binary, so that each document would be either relevant or irrelevant to each query, ignoring repetition of information.

When testing each index system, he would retrieve all documents that system would deem relevant to the query. He would then look at the proportion of returns that were found to be relevant, deeming that value a system's **precision**. Next he would see what proportion of the relevant documents available in the whole collection had been returned, deeming it **recall**. These two statistics would prove fundamental to the Cranfield paradigm and the future of IR evaluation.

In brief, the Cranfield paradigm describes an approach to information retrieval that emphasizes objective statistics derived from the performance of systems in rigorously-defined tests. Precision and recall are the most well-known of these statistics, although their harmonic mean – referred to as the F1 score, F-score or F-measure – is often used to describe overall accuracy.

A typical example of a Cranfield evaluation can be found in Zhang et al.'s *A Comparative Study on Statistical Classification Methods in Relation Extraction* (Zhang, Gao, & Gui, 2013), where five methods for classifying extracted relationships from text are being compared. This comparison involves constructing a test data set of documents properly formatted for the classifiers, derived from a collection originally built for an IR conference. Each competing method's recall, precision and F1 score for each kind of relationship in the data set was calculated and used as both a comparative measure and an absolute performance value. Studies such as this are common, and make up the meat of IR's many test track conferences.

It is worth mentioning that the dominance of the Cranfield paradigm came much later than its inception. Prior to the 1980's information retrieval had yet to reach the large-scale significance it

enjoys today, and calls for a more human-centric approach to evaluation that measured a system's ability to support users pervaded that decade. It would only be in the 1990's, with the rise of conferences like the Message Understanding Conference (MUC) that the Cranfield paradigm would become widespread.

2.1.2 Conferences and Test Tracks

It is impossible to overstate the significance of test tracks to the advancement of information retrieval. A phenomenon not found in every research field, test track conferences provide a competitive environment where researchers work to surpass each other at different retrieval tasks. As the selection and calibration of evaluation methods at these conferences will decide the top contenders, a vast quantity of literature has been produced on the subject.

With the rise of the internet and other major information technology projects, conferences dedicated to improving tools like search engines rapidly gained prominence. The Message Understanding Conferences are a prime example, where the United States' Defense Advanced Research Projects Agency (DARPA) promoted research and set many of the standards and practices still upheld today. The MUC conferences were hotly competitive, and in their search for a solid, objective performance measure they would turn to the Cranfield paradigm(Chinchor & Sundheim, 1993).

The MUC ran from 1987 to 1997, but proved so influential that a number of conferences since have modeled themselves on the test track formula. The Text Retrieval Conference (TREC) and the Text Analysis Conference (TAC) are both examples of National Institute of Standards and Technology (NIST) conferences that developed in MUC's wake, and continue to shape information retrieval as a field (Voorhees & Harman, 2005).

A key element of test-track evaluation found across all of these conferences which can be traced back to Cleverdon himself is the establishment of test collections and the concept of relevance. In order to run tests and derive scores, it proved necessary to develop collections of documents beforehand that had been properly prepared and formatted. This might involve grouping them by topic, removing extraneous headers and other graphical components, and in some cases digitizing them if they began purely as paper copies. Depending on the needs of the test in question, further formatting might be

done to separate sentences, clean out punctuation and so on. News reports and (somewhat ironically) other academic papers have become popular sources for test tracks, as they provide large sets of tidy documents easily organized by subject.

There are many practical advantages to the evaluation method employed in the test-track setup. Test collections, once established, can be reused repeatedly at no extra cost. This allows any number of competing systems to be compared, including later competitors. The entire evaluation method is not labor-intensive, with relevance judgments coming from just a small number of assessors instead of a large body of users. Precision, recall and F1-measure are universally understood and accepted both as comparative measures and representing some absolute value for performance. Collectively, the benefits of the test track approach make for evaluations that are orderly, objective and efficient.

These are advantages we would want to capture for our own method, but they do not exist in a vacuum. Each involves abstractions and assumptions made by Cleverdon and extended by the organizers of the MUC and later conferences. We must gain a better grasp of this underlying philosophy before we can adapt its methods for our own ends.

2.1.3 Philosophy

We have so far examined the 'what' and 'how' of traditional IR evaluation, but we must also be able to answer 'why'. The Cranfield paradigm and the associated miscellanea of orthodox evaluation would not be as widespread as they are if there was no underlying reasoning that could relate their scores and findings to the real world. One common criticism of the Cranfield paradigm is that it is too abstracted from the real use of these systems, but this is not entirely the case. There is a unifying theory at work.

The practice of information retrieval is primarily concerned with technical optimization. Maximizing precision and recall is the aim, with the assumption being that a system which presents all relevant documents (and only those), ranked by the probability of their relevance, has done an optimal job of satisfying a user's query. This is because the *philosophy* of information retrieval has made relevance the measure of all success. All that differentiates a successful system from a failure is how well it can retrieve and present those documents determined by some authority as relevant to a query. Any

ancillary concerns about whether their actual information need was satisfied or what really constitutes relevance are dismissed as issues for other fields, such as HCI.

This certainty springs from somewhat uncertain foundations, that relevance can be a binary, all-encompassing measure for success. Bourlund's *The Concept of Relevance in IR* (Borlund, 2003) lays out the challenge information retrieval faces in the fact that relevance is a complex, multi-dimensional concept that covers all kinds of ways a piece of information may be meaningful to a user. Information retrieval bases its algorithmic approach to evaluation primarily on squashing this complex subject into a binary of "relevant" and thus valuable vs. "irrelevant" and thus without value, yet struggles to account for even basic complications such as when relevant information is being repeated, or when two users – even experts – would disagree about what is relevant. The irony is that many researchers in information retrieval would agree with using relevance and statistics dependent on it like recall and precision but would not agree with each other as to how relevance is exactly defined or why it is of value.

The most penetrating critiques of Cranfield orthodoxy, then, are less how it works but rather what it is used for. In the provocatively-named *The fault, dear researchers, is not in Cranfield, But in our metrics, that they are unrealistic* (Smucker & Clarke, 2012), Smucker and Clarke suggest that the Cranfield approach of systemic, statistic-driven evaluation is a valid approach to predicting a system's performance. The issue is merely what statistics are being measured and what they are interpreted to mean. Their own time-biased gain calculation takes a more user-based approach by measuring predicted gain to the user over time invested, a strong example of how Cranfield principles of rigorous testing can be tuned to other features of a system's performance.

This critique shows why we cannot be satisfied with a standard Cranfield evaluation. While it is a potent paradigm, it is tied very tightly to a particular kind of system and test environment. If we wish to satisfy more sophisticated information needs than the common query, we must be similarly willing to look beyond precision and recall.

2.2 Alternative Evaluation Methods

Not all alternative evaluation methods follow the same reasoning – some, like the aforementioned time-biased gain, stay broadly within the confines of the Cranfield paradigm while introducing specialized metrics of their own. Others go further in developing entire new evaluation philosophies. Looking at how these methods developed can give us insights on our own as well as help determine the scope and size of our deviation.

2.2.1 Evaluation Metrics

It is not always necessary to reinvent the wheel in evaluation. Often, a new problem, system, or perspective on an existing information retrieval approach can be captured purely by new metrics, or even by introducing variants on traditional precision/recall thinking. This has the attractive benefit of being able to justify your evaluation according to an already widely-accepted line of reasoning, so long as you can prove your own version is consistent.

One example is the relatively recent rise of novelty and diversity as important facets of returned results. As noted previously, the Cranfield paradigm's definition of relevance considers each document's relation to a query independent of other documents. This can result in a set of returns filled with repeated information.

In *Novelty and Diversity in Information Retrieval Evaluation* (Clarke, et al., 2008), Clarke et al. ameliorate this issue by devising a diversity metric for measuring the diversity of a set of returned documents. By identifying how many different subjects a system returns, we get a better sense of how a system is supporting the actual user experience. All it takes is a minor shift in perspective to achieve new insight within the existing evaluation framework, setting diversity alongside precision and recall in order to make ‘relevance’ more relevant.

An example metric which goes even further afield is retrievability. Proposed by Azzopardi et al. in *Retrievability: An Evaluation Measure for Higher Order Information Access Tasks* (Azzopardi & Vishwa, 2008), retrievability applies more to the documents and queries than to the system managing them. It aims to measure how easy a relevant document may be recognized by retrieval systems as applying to a query. For example, documents that share keywords with a particular query will

naturally be more retrievable for that query. Recognizing the significance of retrievability can modify the outcome of a system's performance evaluation, such as giving additional weight to a system which can recognize and retrieve relevant yet low retrievability documents.

None of these new metrics stray too far from the familiar Cranfield approach to evaluation, but a rare few attempt to bridge the gap to complex IR. Time-biased gain is one such metric (Smucker & Clarke, 2012a), a method for traditional search-engine style systems which explicitly incorporates user modelling into its performance measurement. By going beyond precision and recall to consider a user's patience, their ability to recognize relevance and the time they will spend actually searching for a solution, time-biased gain has exactly the sort of blend between objective test-and-statistics driven Cranfield evaluation and more nuanced, user-relevant evaluation we would want for our own system.

Even time-biased gain is still only adapted for the current generation of information technology. To understand what alternative methods are being developed for the next generation and prepare ourselves to do the same, we must look at paradigms that wander far from the familiar.

2.2.2 Exploratory Search

An excellent guide to the exploratory search paradigm can be found in White and Roth's *Exploratory Search: Beyond the Query-Response Paradigm* (White & Roth, 2009). In that collection, they describe exploratory search as "an information-seeking problem context that is open-ended, persistent, and multifaceted, and information-seeking processes that are opportunistic, iterative, and multi-tactical." A whole new vocabulary is at work.

It should be immediately obvious that the philosophy represented by Cranfield does not mesh with the goals of exploratory search. Relevance cannot be binary in nature if the user's goal shifts as they explores. Information retrieval systems optimized to perform well under Cranfield conditions are similarly ill-equipped for exploratory search, as traditional document-retrieving web search engines are often dependent on the user knowing what they are looking for and what keyword-rich language to describe it with.

Developing this paradigm and the supporting evaluation methods has required developing new philosophies, new user models. In *Assigning Search Tasks Designed to Elicit Exploratory Search Behaviors* (Wildemuth & Freund, 2012), Wildemuth and Freund describe what researchers seeking to study exploratory search must accommodate, particularly in the design of their studies, in order to elicit the desired response from users. Problems that are open-ended, uncertain and given to interpretation work best with exploratory search systems, and so it follows that a researcher seeking to evaluate performance in this field would have to design tests that involve similar open-ended and uncertain tasks.

Exploratory search has given rise to a multitude of evaluation measures. White et al. in *Evaluating Exploratory Search Systems* (White, Marchionini, & Muresan, 2008) give a quick introduction to five different attempts by different researchers to contribute to exploratory search evaluation. While some methods evaluate individual system types, all shy away from attempting an overall evaluation framework for all exploratory search systems, a task which White et al. admit to be a long-term goal.

Exploratory search is not a finished project, certainly not as complete in an evaluative sense as traditional Cranfield-rooted information retrieval, but it is a powerful expression of information retrieval researchers' desire to go beyond the confines of the old model. Exploratory search is just one example of this revived interest, the Human Computer Interaction and Information Retrieval symposium (or HCIR) is another. With such powerful examples of how to develop a new evaluation method, we can consider ourselves well-equipped to taking on entity-relationship systems.

2.3 Entity-Relationship Systems

Now that we understand the history of evaluation in information retrieval, we should also familiarize ourselves with the new variety of system we seek to evaluate and what problems it presents for existing methods.

2.3.1 The Entity-Relationship Model

Entity-Relationship systems are but one species of what we termed complex information retrieval systems in our introduction, but they encapsulate the challenge complex IR presents to traditional evaluation approaches. Their way of connecting related pieces of information together supports a far

different kind of user information need, and the assumptions underpinning the traditional approach are not guaranteed to hold.

The entity-relationship paradigm used in information retrieval has several conceptual roots, but perhaps its most significant comes from software engineering. In the 1970s Peter Chen published a paper describing a model for organizing the components of a software project, including both internal elements and the interfaces connecting users (Chen, 1976). His model described each discrete element of a system as an entity, much like a node in a graph, and connected the entities together via binary relationships that function akin to edges.

This model would meld with other information retrieval concepts, such as Joseph Novak's concept maps, also developed in the 70s (Novak, 1977). Concept maps serve a similar purpose of laying out the interconnected parts of an idea, but moved away from strictly representing software engineering components. The entity-relationship model became a flexible paradigm for representing any set of information you might imagine, digesting and organizing documents ranging from news reports to short stories to academic papers. An example of an entity-relationship graph, showing relationships between one entity and two others, can be seen in figure 1.

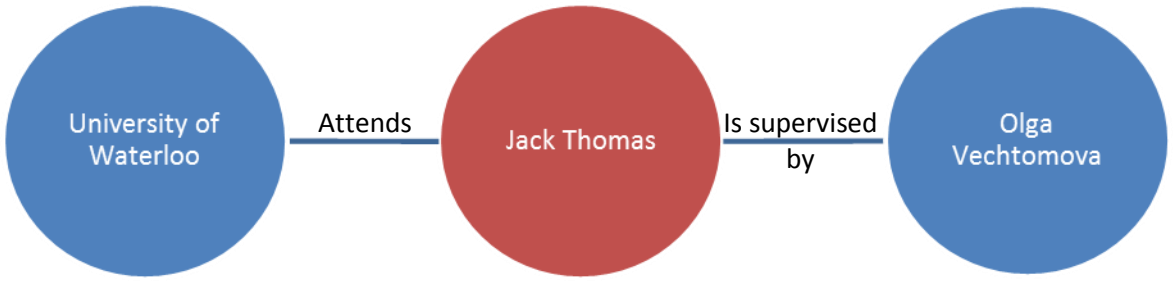

Figure 1: Entity Relationship Graph Example

For much of the 80s and 90s, information retrieval has focused on the more direct goal of document retrieval and search engines. As time has worn on and more complex information retrieval systems have been sought, however, the entity-relationship model has seen a variety of implementations. One breed of system is concerned with taking a subset of entities as an input and building the most informative subgraph out of the relationships that connect them.

This variety of entity relationship system already has two implementations. The earlier of the two, CEPS (short for "centerpiece subgraph system") was put out by Hanghang Tong and Christos Faloutsos in 2006 (Tong & Faloutsos, 2006). It takes a set of input entities and looks for relationships those entities have in common with other entities and with each other, extracting relationships it deems significant. Kasneci et al. would in 2009 put out their own system, MING (Mining Informative Graphs), to perform effectively the same task using an "interestingness" measure (Kasneci, Elbassuoni, & Weikum, 2009).

One feature shared by both of these systems is a weak approach to evaluation. With CEPS, the authors invent a measure for "goodness" built around the specific domain of their test, namely authors and papers, where authors who have written many papers with many coauthors are valued more highly. While an interesting idea, insufficient effort is made to relate this value to real world performance for users, or to explain how the approach could be adapted to other domains and what underlying evaluation philosophy is at work.

MING, coming after CEPS, evaluates by comparison to CEPS. Their evaluation consists only of having test users perform tasks with both systems and then asking them which they preferred, an extremely simplistic approach that at best only establishes a user's preference for one system over the other. There would be no way to compare a third system without conducting another expensive user test, and there is little to no indication of each system's absolute performance or the utility offered to users. There might even be questions about whether the test administered was unbiased, rather than playing to the strengths of MING.

This evaluation problem appears in other varieties of entity-relationship system. ER-FS is a browser-like tool meant to allow users to explore a graph of entities and relationships in a manner not unlike a

classic query-based search engine (Yogev, Roitman, Carmel, & Zwerdling, 2012), but its evaluation is based entirely on the function of the query tool using traditional Cranfield precision/recall metrics. The value of the entity-relationship graph for users and its effect on search behavior passes mostly without comment.

Some varieties of entity-relationship systems have proven receptive to adapting evaluation standards from traditional approaches. Blanco et al.'s *Entity Search Evaluation over Structured Web Data* proposed an entity-relationship search engine very similar to a classic web search engine, using crowd-sourced relevance judgments and real-world queries to create a search track allowing for the comparison of many systems and the production of statistical performance scores.

Blanco et al.'s work is an example of one way to create evaluation methods for complex IR tasks within the entity-relationship paradigm, incorporating both the advantages of the rigorous test track environment and the human factors of crowd-sourced relevance assessments. This was partly enabled by their system being quite close to a classic search engine, meaning we cannot always rely on having an obvious and intuitive adaptation option.

What this should point to is a promising new field being held back by an evaluation bottleneck. Without a coherent, widely-accepted standard of evaluation it would be difficult to hold a TREC-style conference track for any breed of entity-relationship system. While there may not be a single method that could work for all varieties of entity-relationship system, a method for one of them might help uncover common principles useful to others.

2.3.2 Entity-Relationship Recommender Systems

Beyond their organizational paradigm, entity-relationship systems are a diverse lot. Different varieties of system have attracted more or less attention so far. Some have even attracted attempts to create a more standardized approach to evaluation. In order to give ourselves a blank slate to work with, our chosen style of system will be one without any prior evaluation history to reference.

Our example entity-relationship system shall be the **entity-relationship recommender system** (referred to here at times as **recommender system**). A recommender system works with an entity-

relationship graph of a certain subject. This graph may be populated by extracting entities and the relationships between them via an automatic parsing of documents or it may be manually constructed, what matters is that it codifies a topic into entity-nodes and connects them with relationship-edges.

Instead of presenting this graph to the user directly, users are given a starting subgraph, perhaps as little as a single entity-node. When selected, each entity presents a ranked list of its relationships to other entities in the graph according to the system's judgment of what is relevant. Selecting one of these relationships will add the connecting entity to the user's subgraph, and in this way the user can explore a graph by building their own set of entities and relationships of interest to them.

For a graphical illustration of how an entity-relationship recommender system works, look to figure 2.

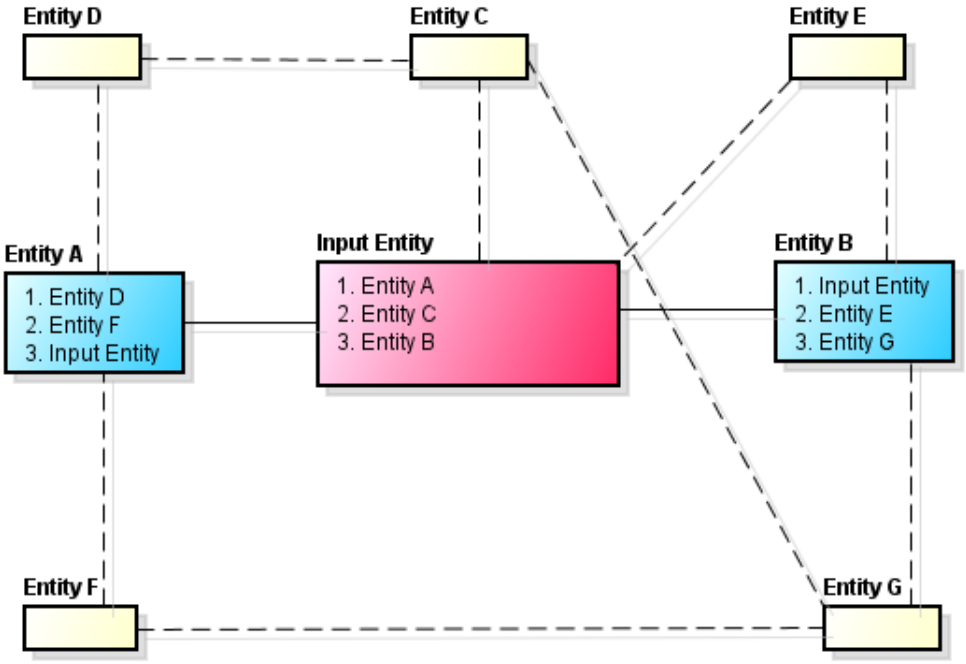

Figure 2: An entity-relationship recommender system in action.

Here we can see an example of a subgraph being built by a user. The user has been given the Input Entity to start with, which recommends relationships to entities A, C and B. The user has subsequently added Entity A and Entity B, and gained access to their own lists of relationship

recommendations. In this way they can continue to build their own subgraph, discovering useful relationships and adding them as they go along.

For a more active example, imagine a user interested in mountain climbing but uncertain where to start learning about it. From a "Mountaineering" entity, the system might recommend relationships such as "Mount Everest is the most famous destination for mountaineering" or "Sherpas are expert mountain climbers". As the user selects those relationships that interest them, their subgraph of available entities grows. They gain an appreciation for the different parts of mountain climbing, with a chance to discover unexpected associations and entities.

This is an activity consistent with the paradigm put forward by exploratory search, where the goal is to support a user between each query on their way to satisfying a higher level information need. While it is similar in spirit and intent, however, in exact implementation it does not fit with existing evaluation methods. Exploratory search may grant us useful insight and a guide to designing new approaches to evaluation, but it remains up to us to develop a working method from this.

The advantage of being a relatively new approach with few current implementations and no organized attempt at evaluation is wide latitude to work. This system is not wholly hypothetical. A company based in New Zealand and Canada named InsightNG[1] is presently working on a commercial software implementation of an entity-relationship recommender model meant to help users pursue new interests. Research projects in the University of Waterloo's information retrieval groups are at this time exploring its potential. A new evaluation method could attract further interest to the subject.

This chapter has left no question about the size of the task ahead. Developing an evaluation method for entity-relationship recommender systems, one which fits into the historical context of evaluation in information retrieval, is no easy undertaking. However, with a strong foundation in the form of the Cranfield paradigm to build upon and useful examples of new evaluation approaches to draw from, we can at least say that we are well-equipped for the task. So we enter the next chapter and begin to

[1] http://www.insightng.com/

tackle the question of what makes a good entity relationship recommender system by going right to the source – the very users we would hope to support with one.

Chapter 3
User Study

As we have seen in the previous chapter, one of the key concerns for evaluation is how the results of said evaluation relate to the experience of real-world users. If an information retrieval system can achieve top marks in evaluation yet fail to achieve its purpose in practice, the evaluation system itself is at fault. Further, most evaluation methods are grounded in assumptions about user's behavior and interests, assumptions which are difficult to make in newer systems. Therefore, to better inform our own evaluation approach, it proves necessary to conduct a study with an actual recommender system.

3.1 Purpose

Before we can get into describing our study, we must explain the exact reasoning that brought it about. Studies can be expensive, time-consuming and risky endeavours, not to be undertaken lightly. We need a motive to match the cost, as well as clearly-outlined goals we intend to achieve.

3.1.1 Motivation

Our overarching motive in launching this user study is simple – we want to know more about entity-relationship systems. How they work in practice, how people interact with them, how to meaningfully evaluate them in relation to the real world. How does a user study fit into that?

A user study lets us put real people in front of an entity-relationship recommender system and see what they do. It allows us to test the hypothetical system itself, to see if people will use it the way we expect them to and if it actually helps them achieve some sort of goal. It allows us to gather a variety of interaction data and see what unexpected patterns emerge in the analysis to better inform our methods.

To measure user satisfaction, we must have users perform the same role as an assessor for a test collection, making relevance judgments. We must give them an objective meant to simulate a higher order information need, the sort of information need which differentiates complex IR tasks from more straightforward traditional IR queries. By gathering users' own assessments of system performance

we can figure out what constitutes success for a system and thus what our evaluation should measure. We can even use these assessments to validate our own evaluation method's results.

Our motivation is, in sum, exploratory. While there are things we wish to see, the potential for unexpected insights and discovery is an equally strong motivator. The dearth of data about actual entity relationship systems is one of the largest obstacles to its advancement as a field, so carrying out a study to shed some light on this new approach is warranted.

3.1.2 Goals

With our motives clarified, then, we should lay out a clear set of goals to be achieved by the study. Our goals will fall into one of two broad categories which eagle-eyed readers will have noted are also the titles of two chapters of this thesis. The first set is behavior goals, concerning the construction of a new user model for entity relationship systems that goes beyond the assumptions inherent in the traditional IR approach. The second set is evaluation goals, concerning the collection of data that can be used to build our evaluation method.

One of our main behavior goals is to test parts of traditional IR theory with an entity relationship system and see what holds up. One popular principle, for example, is that users have a natural preference for whatever is presented at the top of a list – so much so, that even if you take a ranked set of documents and reverse the order, users will follow up on the first few results anyway rather than reading to the bottom of the list (Joachims, 2002).

Another theory in traditional IR is articulated in Russell-Rose and Tate's *Designing the Search Experience: The Information Architecture of Discovery* (Russell-Rose & Tate, 2012). It presents the idea that user search patterns with search engines follow one of four strategies depending on their domain and technical expertise. Technical experts in entity-relationship graphs are fairly rare at this point, as the paradigm is not widely familiar to the public, as such we can focus on the two strategies corresponding to technical novices based on their domain knowledge.

According to the theory, technical novices with expertise in their query's domain will exhibit "depth-first" search behavior where they tend to follow one lead from a set of search results wherever it takes

them, only turning back if they reach a dead end. A technical novice who is also a novice with the subject of their query will adopt a "breadth-first" approach and make many shallow explorations from their initial set of search results, being reluctant to wander far in a topic they are unfamiliar with. This is a theory we could adapt and test with entity relationship recommenders, to see if the same search patterns appear.

As for our evaluation goals, the first goal would be to collect users' relevance assessments of how useful the system was in meeting their information need. As we discussed in the previous chapter, an entity relationship recommender system's main way of supporting users is recommending relationships between entities that are relevant to their interests. As such, having users judge the relevance of the relationships they find is a way to measure how many useful relationships the system was able to provide them with.

It isn't enough for users to interact with just one entity-relationship recommender system. If we want to perform an evaluation, we should really get user relevance judgments for the output of several systems. That way, we can see what differentiates successful outputs from unsuccessful ones, and thus good systems from bad ones. It might also help if users complete more than one task, to give us broader insight and check that trends that hold in one hold in another.

Both of our sets of goals call for our study to involve putting users to the test with an actual entity-relationship recommender system, having them explore a graph, build their own subgraph and assess the relevance of the relationships they end up finding. There are many fine details that must be worked out for our study to proceed, however, which leads us into our methodology.

3.2 Methodology

3.2.1 Design

Our study centers on a pair of **scenarios**, each of which represents one instance of a user with an information need interacting with an Entity-Relationship Recommender System. Each scenario, therefore, has a **domain** or subject, as well as an **objective** which describes the user's goal within that particular scenario. Each complete scenario can be imagined as a discrete test collection as you might

see at a test track conference, with many competing systems attempting it using a fixed set of parameters and inputs.

The domains for our two scenarios are chemistry and the Canadian parliament. Both were chosen for being subjects where one can expect a wide range of familiarity among a Canadian student population (our most likely recruits), as well as being sufficiently dry to avoid controversy. The objectives chosen were "Find relationships that describe ionic bonds between elements" for chemistry and "Find connections between members of parliament" for the Canadian parliament. These were chosen such that unfamiliar users might still have a chance at completing them but expert users would have a noticeable edge. Users will be asked to rate their knowledge of these subjects at the start of the test from one to five (with accompanying explanations of each level) to help us in our later analysis.

During a test track, competing entity-relationship recommender systems would each try to organize entities and relationships concerning these scenarios into **maps**. A map is a graph of entity-nodes connected by relationship-edges. For our purposes at least, the set of entities will remain the same between maps (representing part of the input to the system from the scenario). Systems' maps are differentiated by their choice of relationships to present to each user and what order they rank them in.

For the purpose of our user study, we will have five **grades** of map for each scenario, representing the output of five different systems. These grades will range from Very Good, Good, Neutral, Bad and Very Bad. We will also fix the number of entities per scenario at thirty and the number of ranked relationships displayed for each of these entities at five, which should provide users with enough choices to do some exploration without running through their patience.

It should be mentioned at this juncture that all maps used here were hand-made explicitly for the study. Some thought was given to using entity and relationship extractors to populate the initial graphs, but time, resources and concern for the quality of their output discouraged this. We would not want our evaluation to be affected by how well an unrelated parser did its job, after all.

The fundamental activity of the test will resemble the outline of entity-relationship recommender systems given in chapter 2. Firstly, users are asked to rate their familiarity with the domain of the scenario from one to five (with accompanying explanations for each level). Users will then be presented with a starting entity and five ranked relationships between it and other entities. Clicking one of these relationships will add the corresponding entity to the user's subgraph and give them access to its five relationships and the entities connected to them. After adding five entities to their subgraph, the test will end and the user will be presented with a list of all of the relationships attached to the entities they added. Their final task is to check off all relationships on that list that they think are relevant to their objective. For a screenshot of the program in action, see figure 3 below.

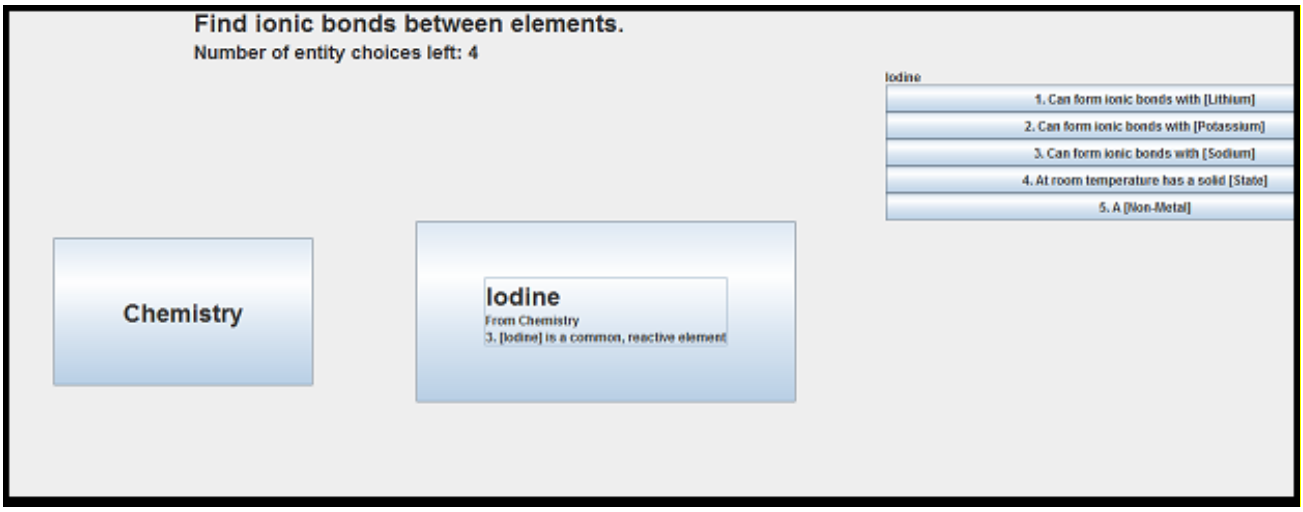

Figure 3: A cropped screenshot of the test program.

Tables 1 through 5 show the basic template used when designing each scenario's five grades of map. Our thirty entities are evenly divided into one of three categories, designated as high, low and mid value. The top five relationships for each entity type will connect to an entity of the designated category – for example, on the Grade 1 map, a mid-value entity's first-ranked relationship will be one connecting it to a high-value entity, and clicking that relationship will add that corresponding high-value entity to the user's growing subgraph.

Of the relationship types shown, only scoring relationships (relationships connecting two high-value entities) are considered to be "actually relevant" for the purpose of our evaluation (more on this in chapter five).

Relationship Rank	High-Value Entity	Mid-Value Entity	Low-Value Entity
1	Scoring Relationship	High-Value Relationship	Mid-Value Relationship
2	Scoring Relationship	High-Value Relationship	Mid-Value Relationship
3	Scoring Relationship	High-Value Relationship	Mid-Value Relationship
4	Mid-Value Relationship	Mid-Value Relationship	Low-Value Relationship
5	Mid-Value Relationship	Low-Value Relationship	Low-Value Relationship

Table 1: Grade 1 - Very Good scenario template

Relationship Rank	High-Value Entity	Mid-Value Entity	Low-Value Entity
1	Scoring Relationship	High-Value Relationship	Mid-Value Relationship
2	Mid-Value Relationship	Mid-Value Relationship	Low-Value Relationship
3	Scoring Relationship	High-Value Relationship	Mid-Value Relationship
4	Mid-Value Relationship	Low-Value Relationship	Mid-Value Relationship
5	Low-Value Relationship	Mid-Value Relationship	Low-Value Relationship

Table 2: Grade 2 - Good scenario template

Relationship Rank	High-Value Entity	Mid-Value Entity	Low-Value Entity
1	Mid-Value Relationship	Mid-Value Relationship	Low-Value Relationship
2	Scoring Relationship	High-Value Relationship	Mid-Value Relationship
3	Low-Value Relationship	Low-Value Relationship	Low-Value Relationship
4	Mid-Value Relationship	High-Value Relationship	Mid-Value Relationship
5	Scoring Relationship	Mid-Value Relationship	Low-Value Relationship

Table 3: Grade 3 - Neutral scenario template

Relationship Rank	High-Value Entity	Mid-Value Entity	Low-Value Entity
1	Mid-Value Relationship	Low-Value Relationship	Low-Value Relationship
2	Mid-Value Relationship	Mid-Value Relationship	Low-Value Relationship
3	Scoring Relationship	Mid-Value Relationship	Mid-Value Relationship
4	Low-Value Relationship	High-Value Relationship	Low-Value Relationship
5	Scoring Relationship	High-Value Relationship	Mid-Value Relationship

Table 4: Grade 4 - Bad scenario template

Relationship Rank	High-Value Entity	Mid-Value Entity	Low-Value Entity
1	Low-Value Relationship	Low-Value Relationship	Low-Value Relationship
2	Mid-Value Relationship	Low-Value Relationship	Low-Value Relationship
3	Mid-Value Relationship	Low-Value Relationship	Low-Value Relationship
4	Scoring Relationship	Mid-Value Relationship	Mid-Value Relationship
5	Scoring Relationship	High-Value Relationship	Mid-Value Relationship

Table 5: Grade 5 - Very Bad scenario template

As you can see, the higher-quality grades work to funnel users toward scoring relationships, while the lower-quality ones try to funnel them away. Tracing the impact this has on how many relevant relationships a user is able to find and correctly identify is a large part of the evaluation significance of the study.

The test regimen followed a standard latin square approach. Each user completes one test from each scenario, with the order and grade regularly varied so that all permutations will be attempted with sufficient participants. Five grades of scenario makes for fifty permutations in our latin square, so fifty users would be needed to complete it (meaning each grade for each scenario would be completed ten times).

3.2.2 Recruitment

The bound of practicality on recruiting a large number of users is one of the reasons for developing this alternative evaluation approach in the first place. The cost in research funding and time as well as the logistical constraints of administering the tests and collecting their results were effectively the limiting factor on most every parameter of the study. That in mind, it was thought that fifty participants, each completing one test from each scenario, would provide a meaningful ten-person sample for each grade of each scenario without exhausting a user's patience and affecting their performance. This also allowed for completing a full latin square, to evenly distribute any potential order bias.

The study's participants were recruited in and around the university using posters, emails and word of mouth. Each participant was compensated with five dollars for their time. As such, participants were predominantly graduate and undergraduate students. Those who had completed earlier, pilot versions of the study were not asked to complete the final version to avoid potential bias from being familiar with the test's structure and content.

The composition of the test group played a role in the selection of scenario subjects and objectives. Chemistry and the Canadian Parliament were both thought to be subjects where students would show a range of familiarity, as some may have formally studied them at university or have a private interest while others would have no interest at all. Similarly, the objectives for both were chosen to be easy

enough to understand for all participants, but those with additional knowledge may gain some advantage when exploring the graph and performing the assessments.

While participants represented a reasonable number of both men and women as well as a number of national and racial backgrounds no official demographic information was recorded as part of the study. This information was thought to be beyond the scope of the research and excluding it would make participants feel more at ease about their privacy and anonymity.

3.2.3 Data Collection

The hard data our test system collects derives solely from the users' interactions with the system. Demographic information, personal preferences and so on fall outside of the scope of our study, although future studies could reveal some relevant insights. Nevertheless the study which seeks to explain everything at once will explain nothing, and so each point of data is tied back to our goals. Every test completed by a user captures the following in an output file:

1. **User's Self-Assessed Domain Knowledge:** A rating from 1 to 5 of how familiar the user in question is with the subject of the scenario.

2. **Choice Type:** Referring to the breadth vs. depth theory, there are values stored for the number of choices a user makes of each type.

3. **User Marked Relevant, Assessor Marked Relevant:** The number of relationships which both the user and the assessor marked as relevant.

4. **User Marked Relevant, Assessor Did Not Mark Relevant:** The number of relationships which the user marked as relevant while the assessor did not.

5. **User Did Not Mark Relevant, Assessor Marked Relevant:** The number of relationships which the assessor marked as relevant which the user did not.

6. **Total Relevant Relationships Found:** The number of relationships among those found by the user which were marked by the assessor as relevant.

7. **Total Relevant Relationships on Map:** The number of relationships the assessor marked as relevant across the whole map.

8. **Rank of Choices**: The ranked position of each relationship which the user clicked on when adding entities to their subgraph.

9. **Number of Entities Looked At During Test**: How many entities the user added to their subgraph that they also selected at some point during the test, so that their relationships were displayed.

10. **Timestamps:** Time taken by users to make each of their choices and then to complete the relevance assessment.

Each user will generate two such files, one for each scenario they complete. Each file will identify the scenario and grade it came from, as well as which latin square position it was a part of. As mentioned previously each grade of each scenario will be completed by ten different users in order to achieve every latin square permutation for the number of scenarios and grades.

3.3 Outcome

While the actual fruits of the study will be used over the course of the next two chapters in our work on behavior models and evaluation methods, a moment should be taken to comment on the conduct of the study itself and the manner in which the data produced was organized.

3.3.1 Study Implementation

The study described in this chapter was cleared to run by the Office of Research Ethics. It was then carried out in late February in DC 3546 where participants could present themselves at any time to complete the study on a lab computer. The vast bulk of the fifty participants took part between February 24th and February 26th. The quick completion time of the test (most participants only took between five and ten minutes) along with generous compensation and an effective advertising blitz allowed for quick data collection. All in all, the study was carried out smoothly and according to plan, with no issues arising.

While interviews and user observations were not formal parts of the study, informal observation of participants and conversations with participants outside of the lab environment did spur some

thoughts regarding the conduct of the test. These thoughts will be saved for later chapters, but do not represent a core or significant part of the research.

3.3.2 Data Handling

The data collected during the study has been organized into a set of spreadsheets in order to generate the statistics needed for the behavior and evaluation chapters. The output files for each scenario have been grouped in two ways, by their grade and by the user's self-assessed domain knowledge. While dividing the data by grade gives an even distribution of ten files per grade (as expected), dividing by domain is not quite so equitable – only one user assessed their knowledge of a subject as 5/5, and relatively few as 4/5.

The appendices of this thesis are not sizeable enough to store all of the raw data gathered by this thesis, but they do contain graphs of all results derived from them. Future chapters will display only the graphs necessary to support their observations, but these appendices are made available to provide a clearer and more complete picture of the study's results. The original data (properly anonymized, of course) may be inspected in appendix A.

3.3.3 Significance

Before we start using the data in our analysis, discussing what parts of our findings are statistically significant would be worthwhile. With fifty participants having completed both the Chemistry and Parliament scenarios, our figures have enough data supporting them to reasonably conclude that they represent real trends within that scenario. Our analysis focuses on explaining these trends and understanding what the users experienced, in the belief that these explanations can provide us useful insight and help us validate our methods.

What we cannot say is that our findings necessarily generalize across all imaginable scenarios. The two we have used here each represent one case of a user having an information need and using an entity relationship recommender to satisfy it, but it may be that certain needs are different enough that users would behave differently while fulfilling them. Proving this kind of generalization would require a huge number of scenarios, each needing a statistically significant batch of users, which strains the bounds of practicality.

Chapter 4
Behavior

Our study has provided us with a raft of data on how users interact with an actual entity -relationship recommender system, but what does it tell us? How can we use it to support the development of our new evaluation method, or advance research and understanding of entity-relationship systems? For this, we get into the hard data produced from our study, but first a brief examination of user models and the role they play in evaluation.

4.1 Importance of User Behavior Models

User models are key to evaluation throughout information retrieval. As a field defined by how it supports and enables users to explore information and find answers, researchers must understand how users do this and what they need from their systems. Even evaluation approaches like the Cranfield paradigm, which focuses heavily on abstract statistics and tests far removed from individual user experience, incorporates a few basic assumptions about the needs of users.

4.1.1 User Behavior Models in Traditional IR

Some of the behavioral assumptions in the traditional Cranfield paradigm have already been discussed in chapter two. F1-score, precision and recall are all considered useful measures because they fit into a model where users want accurate answers to their individual and short-term information needs. They arise from natural assumptions that the best set of results is one made up only of relevant returns, and that all relevant returns are retrieved.

Unfortunately, they are also now deeply buried in the traditional paradigm. Long-held wisdom about the impact of ranked lists on user perception or different patterns of search behavior are ingrained in current evaluation methodology to the point that they are taken for granted. As such, user behavior as a general concept and findings related to it are generally dismissed as being extraneous to the practice of IR evaluation. The HCIR symposium, the alternative metrics and paradigms, they can all be understood as a reaction to the mainstream approach’s relative indifference to questions of user behavior.

That does not mean there are no models, nor pieces of general wisdom or "common sense" in the field that we might not appropriate. If anything, their absence from regular discussion is a product of how they are taken for granted. Assumptions regarding the impact of ranked lists, search behavior and how to satisfy the user's information need are so ingrained they become invisible. The question is whether we can make use of them ourselves.

4.1.2 Adapting Traditional Models

With some models and theories already in place, it would be helpful if we could transfer some over to our own evaluation approach. This can sometimes be done painlessly and intuitively, when the system in question is similar enough to the IR mainstream – for example, we previously mentioned Blanco et al.'s work(Blanco, Halpin, Herzig, Mika, Pound, & Thompson, 2011), creating an entity search engine closely resembling existing web search engine. Most of the same behavioral assumptions would therefore also hold.

There are ways we could imagine our entity-relationship recommender system as fitting the traditional mold. Each entity is a sort of query, while the corresponding set of ranked relationships is like the set of documents a search engine might return, to which we could apply precision and recall. There are difficulties, however, the first of which being that not everything in a new system will have an easy analogue to a traditional one - the activity of navigating the graph via these relationships, for example.

The fundamental difference between old and new models comes from the fundamentally different kind of user each is designed for. Traditional IR seeks to assist a user with a single, direct information need tied to one well-formulated query. The user in more complex IR is one with a larger need, one made up of smaller queries, often one which the user does not yet understand and will only refine as part of the search process. There are whole strata of behavior there that the traditional methods and models simply aren't looking for. As much as adaptation will be a part of any new behavior model, we must also be prepared to produce new theories whole cloth. That makes now as good a time as any to look at some of that new data.

4.2 Study Findings and Implications

The first step toward building our user model for entity-relationship recommender systems is simple – we just look at the data we have collected and start hunting for interesting patterns. Of the ten varieties of data listed in the previous chapter, several of them stand out for extra attention: Choice type ties into a specific behavior theory from traditional IR, relationships marked relevant by user shows some unexpected attributes and the average ranks of people's relationship choices from ranked lists is a significant statistic in IR. The following subsections examine what these data points reveal to us about users.

One data type that did not prove especially useful or undergo serious analysis was the timestamps. As a researcher was on-hand to administer the tests and could confirm that no one took less than two minutes or more than ten to complete any one scenario, the original purpose of the data – filtering outliers who blew through the test without reading anything – became unnecessary. It was also thought unlikely to correlate to any particularly interesting behavior, especially due to how short the test was designed to be, and could easily be influenced by outside factors such as the presence of others in the room, brief pauses to ask questions about the test and so on.

4.2.1 Breadth vs. Depth Search

An example of a behavior theory in traditional IR we might want to adapt is one Tate and Russell-Rose described in *Designing the Search Experience: The Information Architecture of Discovery* (Russell-Rose & Tate, 2012). We mentioned it during the last chapter, where they suggest users with higher domain knowledge will likely favor depth-first search, while users less familiar with the topic with exhibit breadth-first search. In theory, this is a kind of behavior we could easily adapt to our own systems, as the order in which users can be expected to explore the entity-relationship map would be a significant factor in our evaluation method.

To test if this holds, we asked participants at the start of the test to assess their domain knowledge from a scale of one to five, with accompanying explanations for each level. Then, during the test, we recorded each relationship users clicked and the entity it added to their subgraph. Each of these choices fell into one of three categories:

- **Type A**: The relationship is from the same entity's list as the last relationship chosen by the user, indicating breadth-first search.

- **Type B**: The relationship is from the list of the entity which was added by the user's last choice, indicating depth-first search.

- **Type C**: The relationship is from neither the last nor the newest entity's lists, suggesting a dead end but being rather neutral on the topic of search pattern.

In each case, the first choice the users made was discounted, as they would only have their first list to choose from and thus the choice could not be an A, B or C. That leaves four significant choices to be made. This data was divided according to the self-assessed domain score for each user, which reflects their judgment of how knowledgeable they are about a specific topic from a scale of 1 to 5. The 50 participants per scenario are not evenly divided into 10 per domain the way they were into grades. The distribution of users to domains follows in table 6:

User Domain	Chemistry	Parliament
Domain 1	2	14
Domain 2	6	21
Domain 3	26	11
Domain 4	14	2
Domain 5	0	1

Table 6: Number of participants who self-identified as each level of knowledge, out of 5

As can be plainly seen, the distribution leaves much to be desired. Still, within each scenario there is enough diversity that perhaps some pattern can be seen. What follows are the tables showing how many choices users of each domain knowledge level made of each choice type within each scenario. A **special note on figures**, the data for Figure 4 through 17 and 21 through 24 are organized as box and whisker plots following standard min-Q1-Q3-max with the median as a blue dot. Figures 4 through 11 are organized by domain knowledge, while the rest are organized by scenario grade.

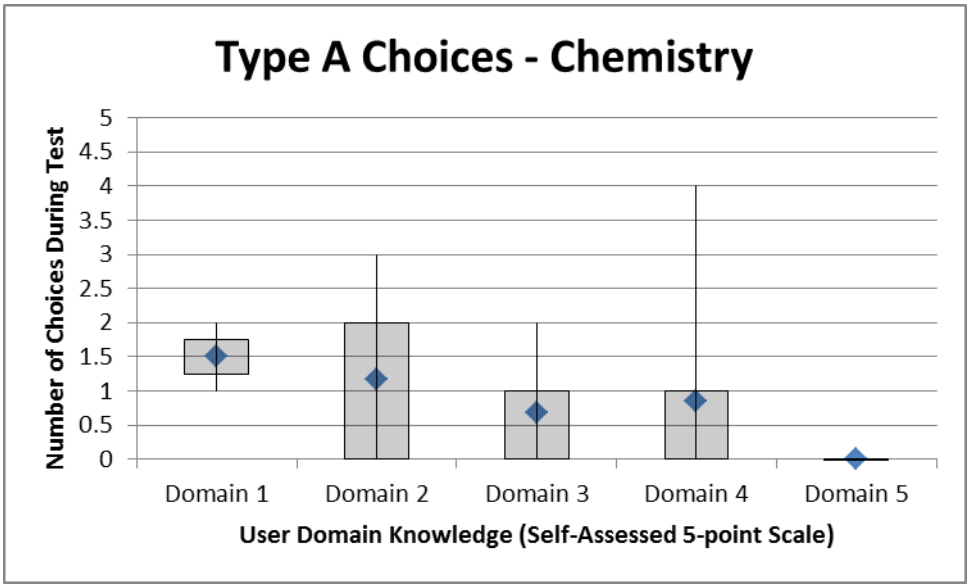

Figure 4: Number of Type A choices for participants completing the Chemistry scenario

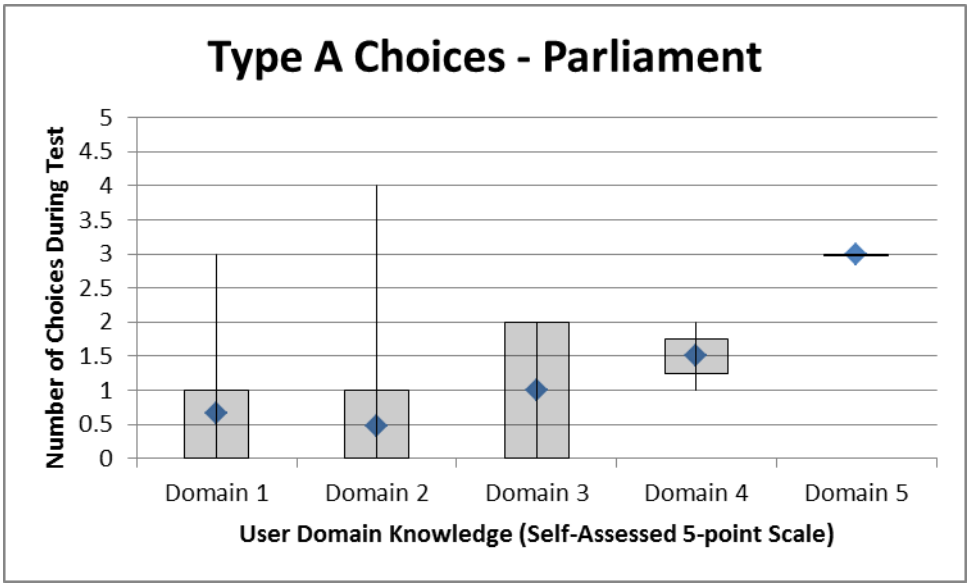

Figure 5: Number of Type A choices for participants completing the Parliament scenario

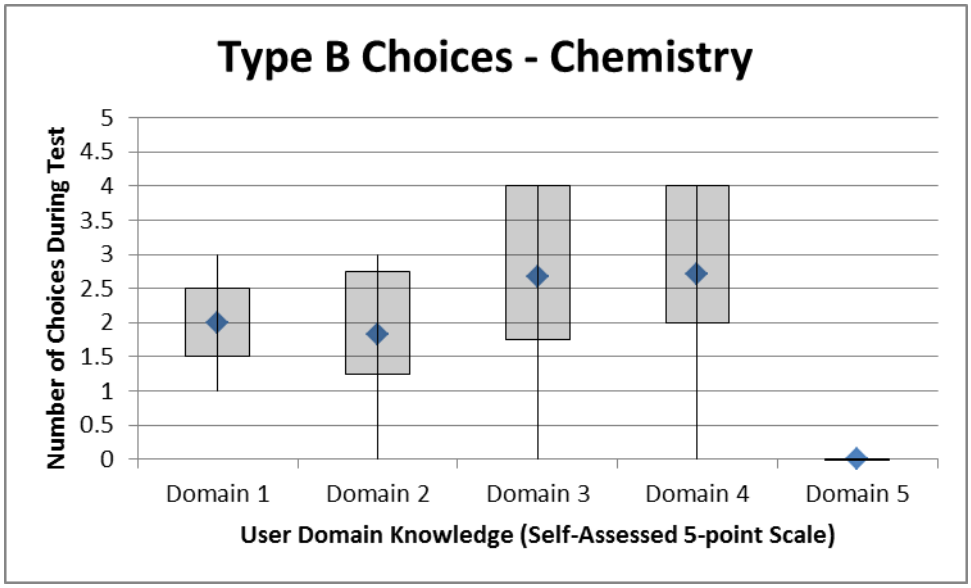

Figure 6: Number of Type B choices for participants completing the Chemistry scenario

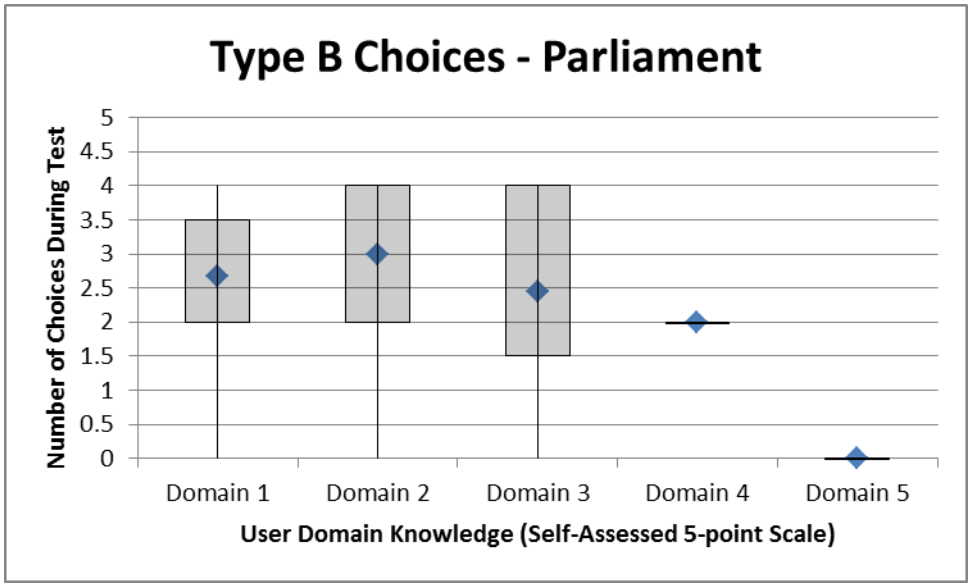

Figure 7: Number of Type B choices for participants completing the Parliament scenario

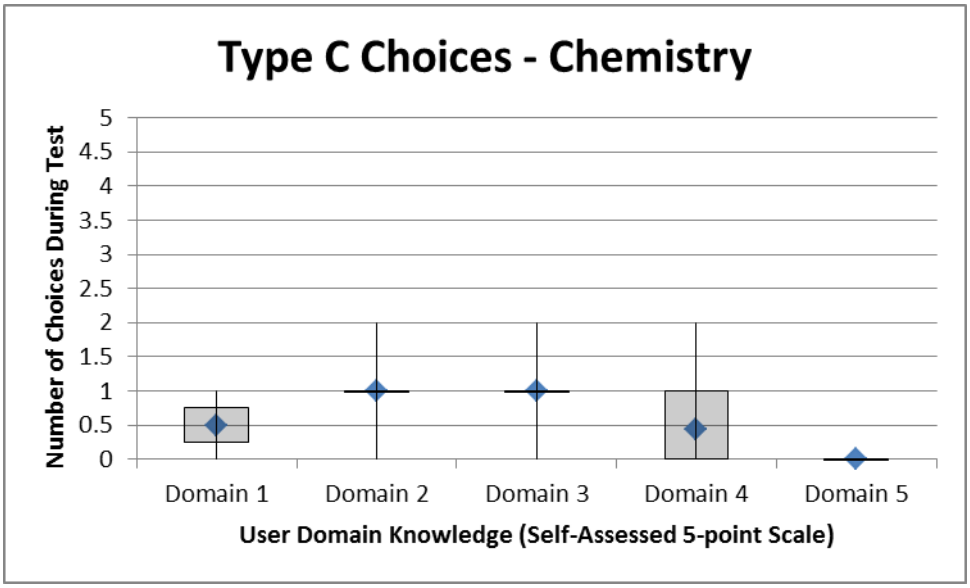

Figure 8: Number of Type C choices for participants completing the Chemistry scenario

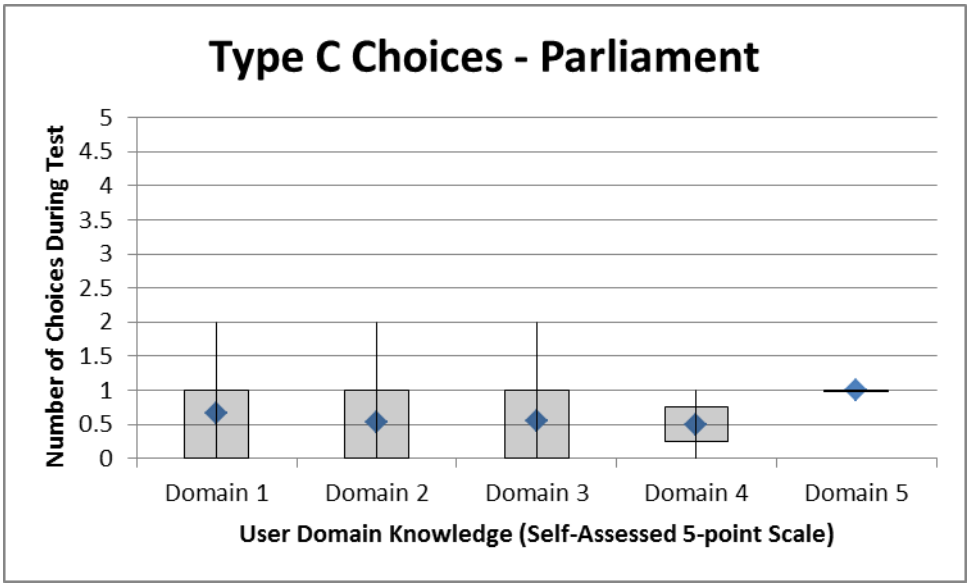

Figure 9: Number of Type C choices for participants completing the Parliament scenario

The trends within each choice type offer little support for the notion that domain knowledge is tied to a user's breadth vs. depth search behavior. Indeed, the one domain-5 user in the whole study

primarily made type A choices, which suggest breadth rather than depth search in direct contrivance to the theory. The more interesting insight the data provides is that across the board, type B appears clearly the most common kind of choice made by users. That would suggest that regardless of the expected familiarity of users with a subject matter, favoring depth over breadth would yield better results both in system design and evaluation methods.

Another supporting piece of data for depth-first search behavior's importance is data collected on how many of the entities users have added to their subgraph that they look at before the end of the test. If users were demonstrating breadth-first behavior, they would likely stay on an entity until all options are exhausted without checking their new options. The data tells a different story:

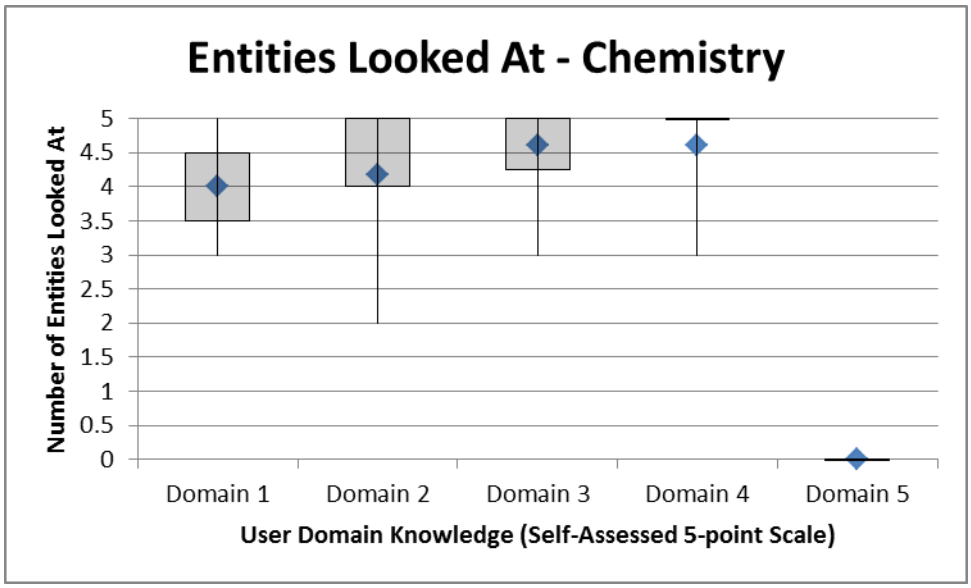

Figure 10: Number of added entities looked at by participants completing the Chemistry scenario

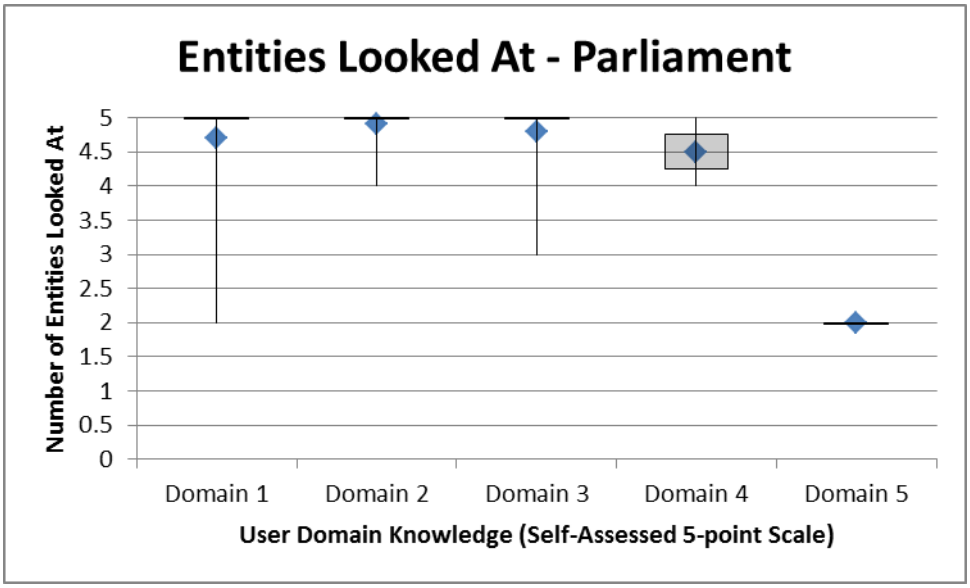

Figure 11: Number added entities looked at by participants completing the Parliament scenario

It might be possible to argue for somewhat of a trend in the chemistry results, but the more meaningful finding is that most users would look at all or almost all of their available relationship options before making decisions, and certainly before the end of the test. This is more indicative of depth-first search, or at least the important feature of depth-first search that users actually check the avenues they open up during exploration.

This also helps to assuage one concern raised during the conduct of the study that asking users to judge the relevance of relationships they found, yet might never actually have seen during the test, is not a fair assessment of what the system has actually taught them. If users are in fact reliably looking at newly-added entities and assessing their new relationship options during the test, this is less of an issue.

Another reason not to be concerned about users not seeing relationships before being asked to judge their relevance is that our system and the evaluation thereof is only meant to measure how well it *supports* learning, with no guarantee that the users will actually internalize what they have been shown. As the saying goes, you can lead a horse to water, but you can't make it drink.

4.2.2 Effects of Ambiguity on Scenario Design

One somewhat unexpected pattern which emerged from the data concerns ambiguity. Specifically, ambiguity about what relationships users consider to be relevant. During the design of the scenarios, it was assumed that those scenario grades with more relationships assessed as relevant (the scoring relationships from our scenario template) would have a higher number of relationships marked relevant by users than the low-quality grades. The following graphs show the number of relationships the study participants marked as relevant at the end of the test, organized according to the grade of scenario they completed, where 1 is Very Good and 5 is Very Bad. It suggests a result quite different than our hypothesis:

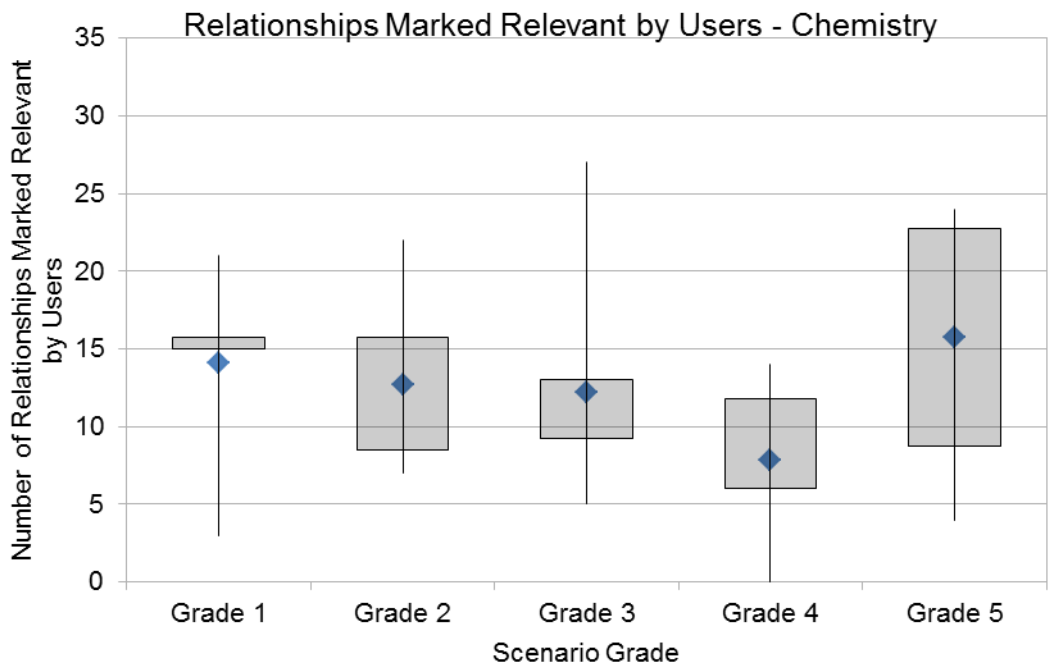

Figure 12: Number of relationships marked relevant by participants completing the Chemistry scenario that were also marked relevant by the assessor

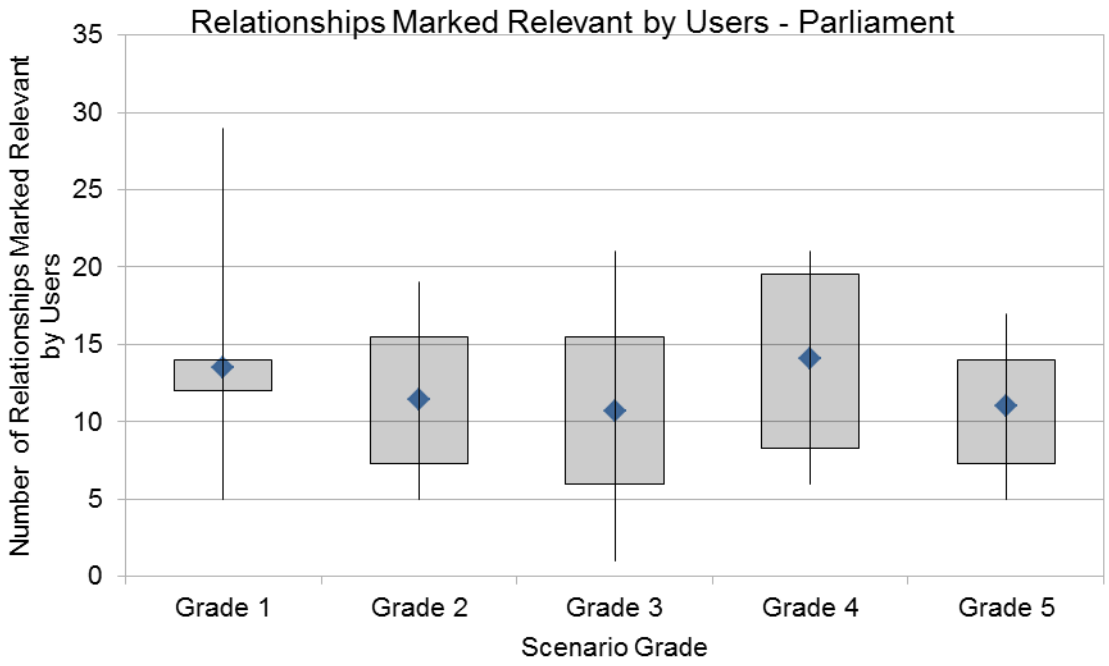

Figure 13: Number of relationships marked relevant by participants completing the Parliament scenario that were also marked relevant by the assessor

This does not mean there is no correlation between scenario grade and what relationships users mark as relevant. Another set of data, showing the proportion of relationships marked relevant by users that were also marked relevant by our assessor, shows a much stronger correlation to the grade of the map:

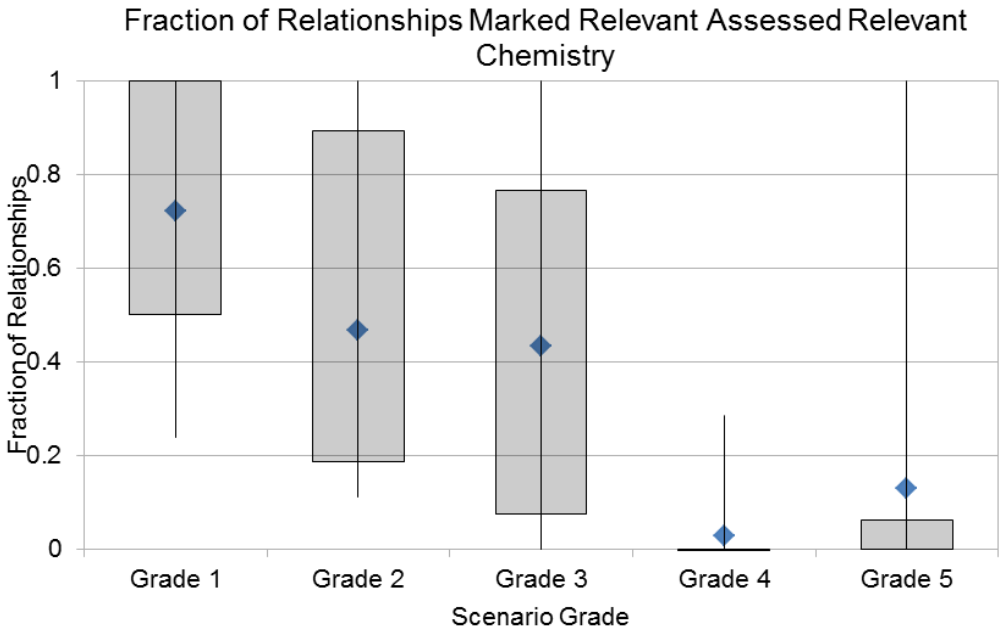

Figure 14: Fraction of relationships marked relevant by participants completing the Chemistry scenario that were also marked relevant by the assessor

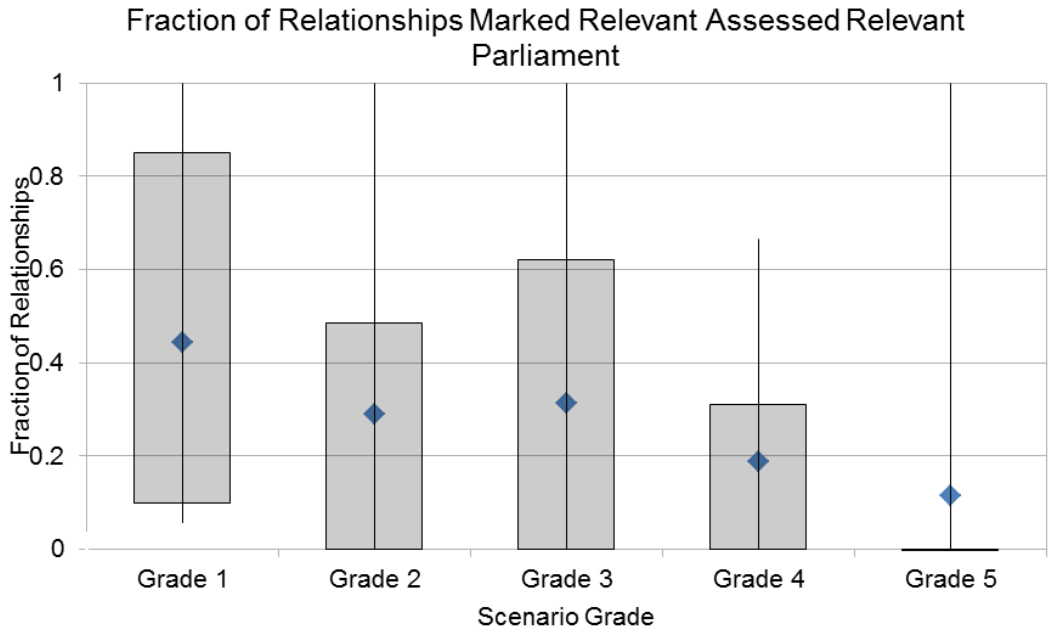

Figure 15: Fraction of relationships marked relevant by participants completing the Parliament scenario that were also marked relevant by the assessor

What this suggests is that users appear to expect a certain number of relevant relationships among the relationships they have found. If users cannot find enough relationships that exactly match what their objective asks of them, they will lower their bar for relevance and include relationships that they might otherwise have excluded. Even during informal observation of participants, some could be seen rereading the list of relationships from the start and checking off some they passed over the first time during their initial sweep.

This sort of behavior is easy enough to understand. An illustrative example of this is giving a student a true or false test where all the answers are true, or a multiple choice test where all the answers are (b). Even if the student knows the right answer, they will begin to doubt themselves. If we had shown all five grades of map to participants first, it might be that they would start to recognize some sets of relationships as more relevant than others and apply more discretion in their judgments.

An interesting little anomaly is the uptick in relationships marked relevant from grade 4 to grade 5 in figure 12. It could be that in the case of a very bad scenario, where users would often find no relationships we had assessed relevant at all, there are no examples of a more relevant relationship so the user's bar is exceptionally lowered. Even finding one relationship that describes an ionic bond between elements on chemistry's bad map might clue a user into the existence of more relevant relationships, but on a very bad map they might think no such relationships exist.

Another possible indicator of this behavior is the much tighter grouping of data for Grade 1 – Very Good scenarios. As these scenarios have a high number of relationships assessed as relevant by our assessor, and because such relationships generally follow a particular grammatical pattern that is quite close to what the scenario's objective describes, it may be that a critical mass is achieved and more users recognize the pattern that relevant relationships take. The widening of the gap from Grade 2 – Good on could be the result of fewer users having this kind of realization and thus a wider range of answers about how many relationships 'should' be relevant.

The specter of ambiguity has sizeable implications for entity-relationship recommender systems and those who seek to evaluate and test them. It evokes Wildemuth and Freund's previously mentioned

paper (Wildemuth & Freund, 2012), where they warn of the importance of test design for researchers in exploratory search. One of the challenges of any system designer is correctly identifying the kind of problem their system is best-suited to solve, and so similarly an evaluator must take care when identifying what kind of problem they will test systems with.

4.2.3 Ranked Lists and their Impact on Behavior

A fairly uncontroversial idea in traditional IR is that for any ranked list, users will be more attracted to the options at the top of the list than the ones near the bottom. In Joachims' *Optimizing Search Engines Using Clickthrough Data* (Joachims, 2002), the significance of the first two results returned by a search engine is paramount. Even if the list is actually ranked in reverse relevance order, if the user doesn't know about the deception they will still show a marked preference for the first options. It's a behavior that search engines have trained the public for over years.

Entity-relationship recommender systems integrate ranked lists in the form of the list of an entity's relationships presented to users. High-quality grades of maps produced by entity relationship recommender systems will rank those relationships from most to least relevant to the user's interests. As such, the same principles may apply.

During the study, data was collected on the rank of each relationship clicked by users while building their subgraphs. As the lower-quality grades of map were designed to rank relationship relevance in reverse order without informing users that this is the case, we are effectively recreating Joachims' experiment.

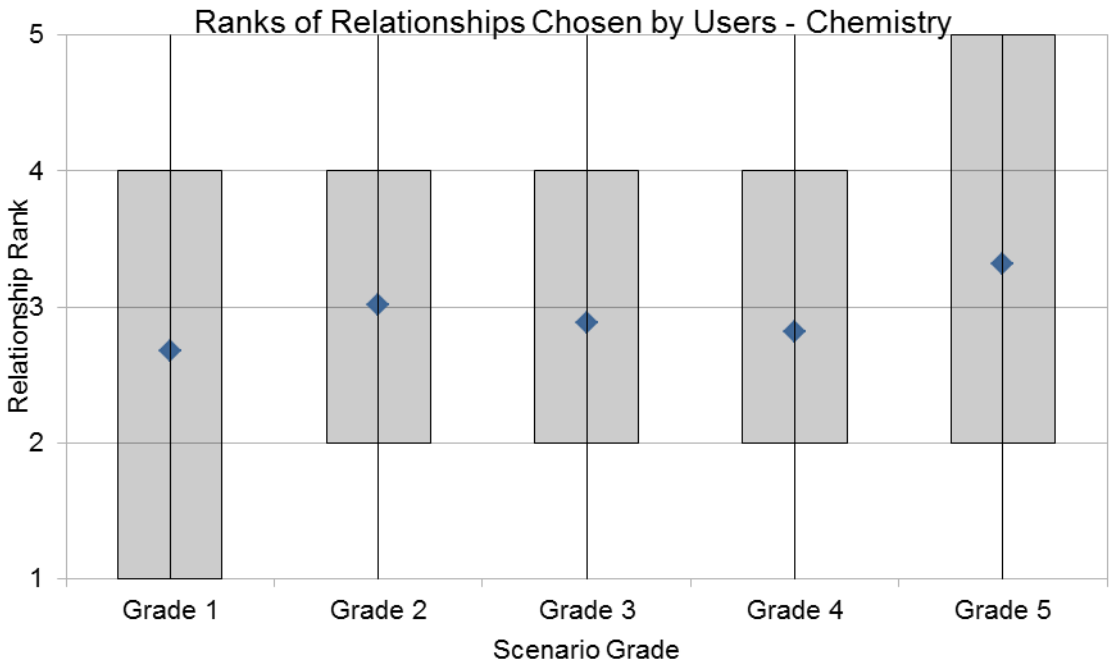

Figure 16: Rank of relationships chosen by participants completing the Chemistry scenario

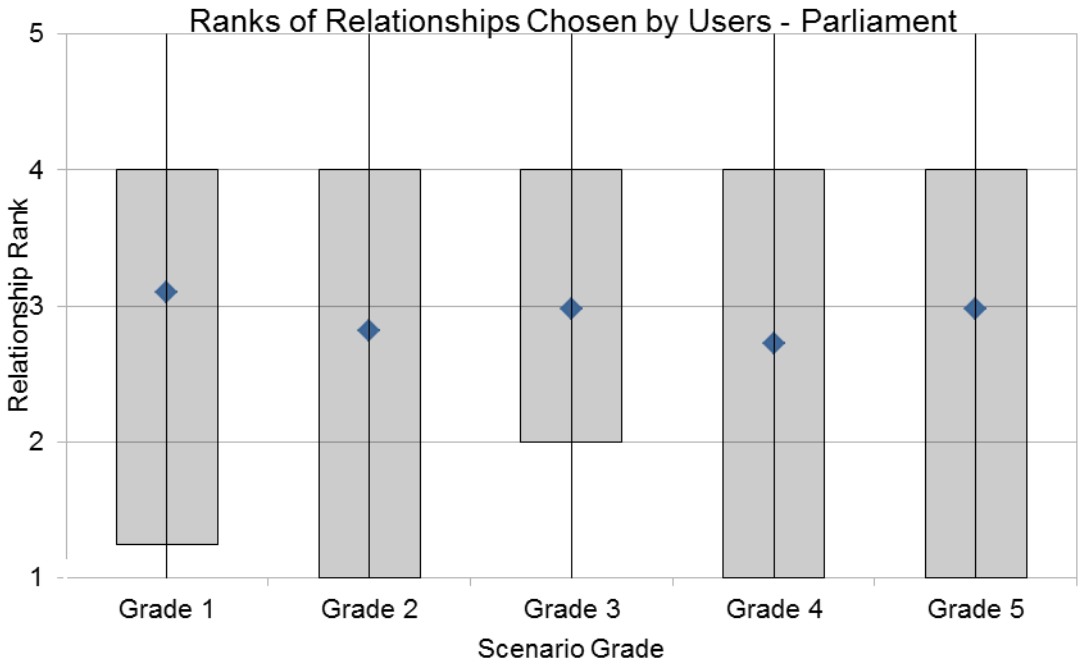

Figure 17: Rank of relationships chosen by participants completing the Parliament scenario

These results have some interesting implications. In chemistry, some effect of the scenario design can be detected, where the users on maps that ranked strongly from most to least relevant have somewhat higher-ranked choices on average than the users on maps that ranked strongly in reverse order. Users on the parliament scenario do not appear to have recognized this trend, and their results more uniformly tilt toward the highest-ranked relationship.

Our previous subsection established the idea of ambiguity, which might be coming into play here. Parliament's trend on figure 15 was weaker than chemistry's on figure 14, already suggesting it was a less clear scenario. If parliament's objective was too ambiguous, it might help explain why users did not notice the reversal here either. They stuck to their default behavior of showing preference to higher-ranked relationships, meaning the power of ranking is still quite strong and active even in entity-relationship systems.

4.2.4 Conclusions

Our data has supported a number of useful conclusions regarding user behavior. For a start, it has helped us figure out what kind of problem entity-relationship recommenders solve best. The effects of ambiguity suggest system performance suffers when users are presented with objectives that are open to interpretation. This does not mean this system is best suited to single, direct queries in the way that traditional search engines are, merely that even a higher-level information need should be clearly defined. A sharper goal, with clearer terms for relevance leads to stronger results – and likely stronger systems, although we will be taking up evaluation in the next chapter. Something to be mindful of when designing test collections, certainly.

Our study data has helped us understand the effect on search behavior that subgraph-building has. Depth-first search has significant implications for designers, such as not being able to rely on users coming back to an earlier list to explore other avenues. It also has implications for evaluation, as evaluation methods that reward systems for supporting depth-first search over breadth-first systems may correlate more strongly with real users' experiences.

The data has also shown that some IR principles still hold. The order of ranked returns is still as significant as ever to users' perceptions of relevance, something we must take note of in any model we would create. That users more or less default to it in cases where what they should be looking for is ambiguous is especially notable, but the fact that it still influences users on very obviously reversed results is still significant. It means we can import behavior theories tied to this concept, and make use of them when we create our evaluation method.

This initial study is just a first dip into the potential of complex IR behavior research. A greater study into all the relevant behavioral aspects of a new IR technique would go beyond what this thesis can cover as a mere sidebar to a larger goal. Our first few deductions, however, can help us immediately. As the next chapter will demonstrate, what we learn here will shape how our method works and what it measures, and user data is a well we will return to when it comes time to validate our new approach.

Chapter 5
Evaluation

Evaluation is the second and greater of our two goals, one only achievable as the culmination of the previous chapters. Only after laying out the history of traditional and alternative information retrieval evaluation, identifying the sort of system we seek to evaluate, gathering data on the needs of users and describing a user behavior model with clear conditions for success can we begin to tackle this challenge.

Our evaluation method for entity relationship recommender systems is covered in three sections. In the first, we shall describe the actual development of the method using the knowledge we have gained from our previous chapters. The second shall be a clear articulation of the method itself and its accompanying evaluation paradigm. The third covers how we may use the results of our study to validate our method's scores and correlate them to the real world.

5.1 Developing our New Approach

The first thing we must do in the development of any evaluation approach is identify the subject of evaluation and the measure of value for a system. For traditional Cranfield approaches, that would be a system's set of retrieved documents in response to a query and the proportion of them that are 'relevant'. The knowledge exploration paradigm takes a broader view, considering not only the relevance of material returned but also how well the system supports the user in developing their larger information needs.

For us, the subject of evaluation is the map of entities and relationships produced by an entity-recommender system, and the measure of value will be the relationships between entities. Each one represents some fact tying two entities together, and collectively they represent the body of relevant returns to be retrieved. The complicating factor is their layout goes beyond merely being ranked in a list, as users must also use them to navigate the graph and grow their own subgraph in order to support exploration.

Our way of accommodating this somewhat ambiguous requirement will be weighting. Weighting the value of retrieved items according to their place in the ranking is already a part of many Cranfield methods, and the user's journey through the graph can be conceived of as a sort of non-deterministic ranked list. Distance from the user's starting position becomes probabilistic, with weights applied to a relationship's value coming to represent their chance of being discovered by the user. These probabilities will be informed by what our study has taught us about user search behavior, both their preference for depth-first search and the influence of ranking.

The variables of entity-relationship recommender systems are the same as they were during the study, and where necessary we will assign them the same values they had during the study. This includes a fixed set of (thirty) entities one of which is designated the starting entity for the user's own subgraph, a limit on the number of relationships that may be displayed per entity (five) and a maximum number of choices the user gets to add entities to their subgraph (also five). Knowing now what form our method must take, we will lay out the details.

5.2 Entity-Relationship Recommender Evaluation

Our method builds from the ground up, starting with the nuts and bolts of individual relationships' relevance to the user's information need. From there, we determine how to score each entity's list of relationships, and then each individual run through the system's weight. After using these to generate a final score for a map, we will then go about normalizing that score to get a clear, meaningful result.

5.2.1 Relationship Evaluation

The first step in evaluating an entity-relationship recommender system's output is assessing the relevance value of all relationships between entities that are part of a given scenario. The actual procedure here can vary according to the requirements of the one conducting the test, much as with relevance assessment at test track conferences, the key point is that a small panel is easier to muster than a large body of users and (as we examine later) can be just as reliable.

For the purpose of this thesis, relevance was assessed by a single assessor in a binary fashion. All relationships relevant to the scenario's objective were assigned one point each – this would cover all relationships marked as "scoring relationships" on the scenario templates in chapter three. All other

relationships were assigned zero points. This means that each entity has a certain number of score-granting relationships associated with it, and how the system chooses and ranks them when producing a map will determine how many points the system will score for that entity.

5.2.2 Scoring a Ranked List

In order to score an entity's ranked list of relationships, we must take into account not only the value of the relationships in the list but also their ranked position. Naturally, a list which has ordered its relationships from most to least valuable is more likely to lead a user down a relevant path than one that presents them in reverse order. This scoring gives us the **ranked list value (RLV)** for an entity according to that particular map.

The RLV is sensitive to a parameter of the scenario, namely the number of ranked relationships each entity may have. Many entities can be expected to have a large number of relationships, certainly more than a user might examine before choosing one. If this parameter, which we will call *R*, is low enough to clip relevant relationships it might make some entities more or less valuable than others. This is a matter of calibration dependent on the scenario designer, for our purposes an *R* of 5 as we used in our study will serve.

Our formula for calculating the RLV uses discounted cumulative gain, a function not unfamiliar to information retrieval research, in order to properly proportion the significance of each ranked position on the list to the overall score:

R = the number of ranked relationships each entity will have according to the scenario
i = rank position of the relationship on the list (up to *R*)
Rel_i = relevance score of nugget/relationship at rank position *i*.
$Log_2(i)$ = discounting factor for relationship's ranked position on the list.

$$RLV = Rel_1 + \sum_{k=2}^{R} Rel_k / \log_2 k \qquad (1)$$

Equation 1: RLV Calculation

One might fairly note that with this particular formula, the value of relevant relationships after the first position drops rapidly. While the first two relationships are added together at full value, the third is weighted at only 0.63, the fourth at 0.5, and the fifth at 0.43. With our system of valuing relevant relationships for one point, that makes the maximum RLV for an entity 3.56, with half of the value coming from the first two relationships.

This ties in to what we learned in the previous chapter, and what Joachims found in *Optimizing Search Engines Using Clickthrough Data* – namely, that when you present a user with a ranked list, it's the first two entries on the list that matter most of all. Users are trained at this point to treat all ranked lists as being ranked by relevance, so the lion's share of the score should depend on whether the system has successfully made the first two relationships relevant.

5.2.3 Runs and Weights

Calculating an RLV for each entity of a map is just part of the puzzle. Beyond having a relevance value, each relationship is also a connection to another entity and controls a user's navigation through the map. An entity with a huge RLV wouldn't do much good if the user doesn't actually reach it, so in order to take this into account we must work out the probability of a user taking each possible "run" through the map and weight the value of that run accordingly.

A run represents the choices one user makes when navigating and building. All runs begin from the same entity, and will involve the same number of choices for adding entities to the subgraph – both parameters of the scenario, and thus will apply to all maps of that scenario being evaluated. Each run corresponds to a potential outcome a user might have with this particular map, so our score will be a function of the relevance value of that outcome and its likelihood.

In order to score a run, we add up all the RLVs for the entities along its route to produce a score for that run's relevance. After that, we weight this score by multiplying it against the probability of the run, a value derived from the rank of the relationships a user would have to select in order to get this run. So for example, a run made of three first-ranked relationships would be more likely, and given a higher weight, than one made up of three fifth-ranked relationships. If the run you get from choosing

the lowest-ranked relationship for each entity is worth more than the one you get by choosing the highest-ranked, your system has done a poor job of ranking relationships!

In order to calculate a run's weight, or *RW*, we use the following formula:

C = number of entities the user selects in a run.
$\mathbf{R_{\#}}$ = The rank position of the relationship on a ranked list when chosen in the run.

$$RW = \left(\frac{1}{R_1} + \frac{1}{R_2} + \cdots + \frac{1}{R_C}\right)/C \qquad (1)$$

Equation 2: Run Weigh Calculation

After that, it's merely a matter of multiplying the *RW* for the run by the total of the RLVs for all entities that are part of that run. For example, a run where the user makes two choices and chooses the first relationship from the first list and then the second relationship from the second will be calculated as (1/1 + 1/2)/2 = 0.75, while one where they choose the second relationship from the first list, the third from the second list, then go back to the first list and pick the first would be (1/2 + 1/3 + 1/1)/3 = 0.611.

An observant reader may note that this weighting system gives preferential treatment to depth-first search. A run where a user only selects the first relationships from three entities will have a weight of (1/1 + 1/1 + 1/1)/3 = 1, while one where the user selects the first three relationships from one entity will have a weight of (1/1 + 1/2 + 1/3)/3 = 0.611. This bias is supported by our findings in the behavior chapter that demonstrated users have a preference for depth-first search.

5.2.4 Calculating the Final Score

The final score for a system, then, is the sum of all of their possible runs. There are a fixed number of possible runs for each scenario, which is a function of the number of relationships each entity has and the number of choices the user makes. Nevertheless, even a small scenario will have so many possible

runs a user could make that working out the score for each one by hand would be infeasible. Thankfully, a fairly simple code script can do this work instead.

To give a simple example, a scenario where each entity has two relationships and the user is allowed two choices will have a run where they select the first relationship of the starting entity then the first relationship of the next entity, the first of the starting and then the second of the next, the first of the starting and then the second of the starting, the second of the starting and then the first of the next, and lastly the second of the starting and the second of the next – five possible runs.

Each of those five runs will have a score made up of combining the RLVs for the two entities added then multiplying the total by the weight for that run. These five run scores are added up, giving us the final score for the map!

5.2.5 Normalization

While it is well and good to have a final score which might be used to compare the performance of multiple systems, it would be better if the score itself meant something. As it stands, the value is an abstraction representing nothing more than the output of a scoring formula. One last step taken to normalize this score against the scenario's top possible score will give a clearer image of how the system performed, not just relative to other systems but in absolute terms.

There are some challenges to calculating a top score for a scenario, rooted in the fact that an entity's relationships are both carriers of value and gateways to other entities. It may be that the ranking of relationships that gives an individual entity its highest possible RLV is not the one that will give a system its highest possible score.

Consider the case of an entity with one relevant relationship connecting to an entity with none, but with one irrelevant relationship that connects to an entity with four relevant relationships. Maximizing our RLV calls for ranking the relevant relationship at the top, but ranking the irrelevant relationship connecting to a valuable entity second will reduce the weight of that valuable entity in the run. Finding the exactly optimal arrangement of relationships in that case would be exceedingly

complex, short of evaluating every single possible map and picking the highest scoring one – a rather processor-heavy proposition!

In order to ease our calculations, then, a simplifying shortcut is taken. The top possible score for a scenario is calculated as though each run were made up solely of the entities with the highest RLVs, regardless of their connections. So, in a scenario like the two in our study with thirty entities and where the user makes five choices, the thirty entities would be listed from highest to lowest RLV and the top five would be chosen as the selected entities for every possible run.

This might result in a run that is impossible for any actual system to achieve, such as combining two entities no fewer than three degrees apart when a user is only given two choices. Certainly, it will result in a run that is wildly improbable, especially as the result for all possible sets of choices the user could make. That the top score we will normalize our systems' performance against may not actually be achievable is acceptable. It represents an upper bound of the relevant information a particular set of entities and relationships has *available*, and scores normalized against it are measuring how close they come to an ideal layout.

Normalization is the final step to our evaluation method. For a step-by-step summary of the method, refer to figure 18.

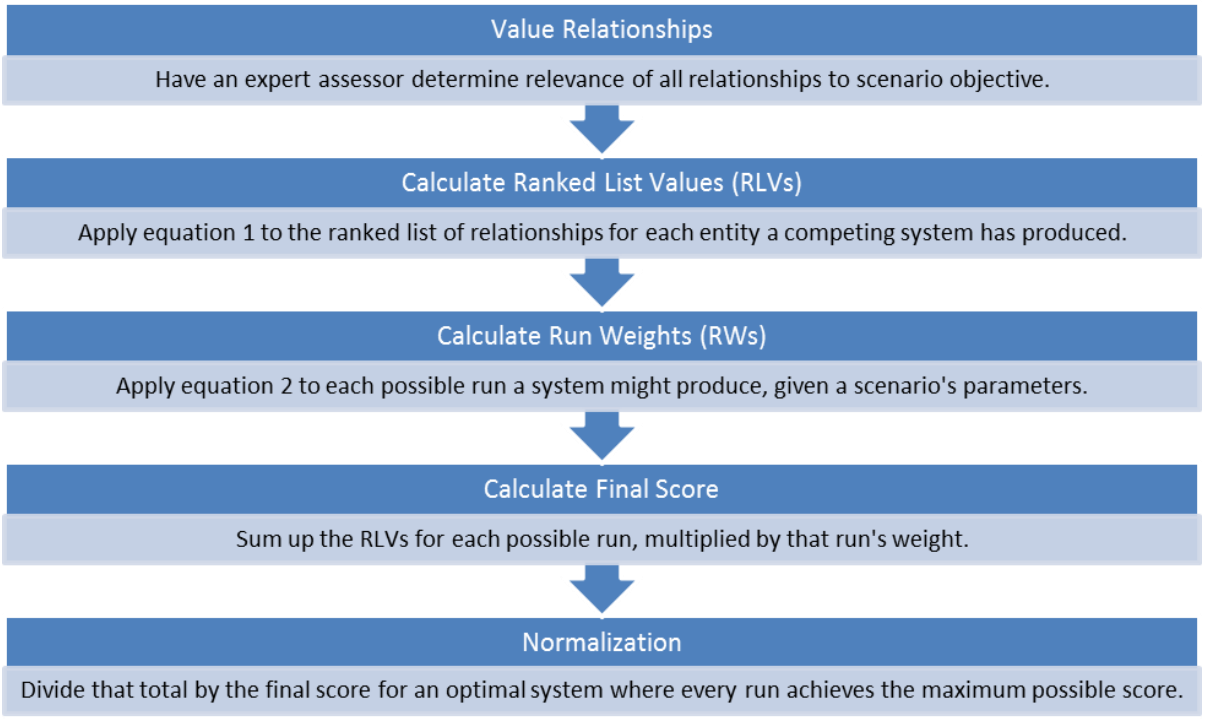

Figure 18: Evaluation Method Summary

5.2.6 The Paradigm at Play

After spending so long discussing evaluation philosophies earlier, it behooves us to explain the underlying principles at work with our method.

Unlike the classic Cranfield evaluation, our method is not targeting precision and recall – at least, not directly. True, similar means are being employed in that we are valuing ranked lists of returns, but precision and recall are rooted in the idea of answering a single question. In order to capture an evolving information need made up of smaller goals, we have developed a method aimed at valuing both the information in the graph and its distribution. Our score is sensitive to how a system performs at recognizing relevant information and at recognizing optimal paths to that information.

Likewise, it has avoided the pitfalls of some of the more vague evaluation methods to come out of alternative paradigms. It is not bound to a single domain, as any topic can produce a test collection with the right set of facts. It also does not require applying a large number of users, although its

prescriptions do track with their experience. Our scores have an explicit meaning rather than being an abstraction useful only for comparison, and we are not limited to comparing two systems at a time.

The advantages of our approach is that it captures many of the benefits of the traditional Cranfield paradigm – namely, it produces a firm percentage score easily intelligible to a lay person, it can be calculated entirely automatically with the application of the right inputs, it requires only a small number of expert assessors instead of a large body of users and it enables comparison between a large number of systems. However, it also achieves the goal of alternative evaluation paradigms like exploratory search by representing the system's performance at a higher-level task.

Our paradigm is sound, but like the other evaluation approaches it has in-built assumptions based on user behavior. We exhibit a slight bias for depth first search and prioritize the importance of ranked lists, both assumptions supported by our behavior data. We acknowledge the effect of ambiguity on our evaluation's results and urge that test collections account for this in their design. The most important assumption, the one underpinning the entire method, is that thanks to our systemic approach to scenario grade design, our scores are indicative of how much available information each system has made accessible to users. This assumption remains to be proven.

5.3 User Study Validation

Having a method of evaluation is a good start, but to establish that its pronouncements have meaning is another matter. In order to validate our method, we must show it correlates to real-world user satisfaction. Fortunately, our study has provided us with more than enough data on that front.

5.3.1 Validating via User Study

The idea has been put forward in this thesis that collecting user relevance judgments can allow us to validate an evaluation method for entity relationship recommender systems, and now is the time to elaborate. Our reasoning on this point is fairly straightforward.

After being normalized, the scores our evaluation method produce purport to represent the fraction of relevant information available in a scenario that a map has made accessible to users. If this is the case, then users interacting with those maps should find more relevant relationships when building their

subgraphs and assess more of their finished subgraph's relationships as being relevant. As these are both statistics we have gathered during our study that means we can use our method to score the maps used during the study and compare those scores against our user relevance judgments to see if our expected correlation emerges.

5.3.2 Evaluation Scores for our Study's Maps

Both of our scenarios from the study had five maps, corresponding to five grades of quality. If our scoring method functions as intended, running the five maps should generate scores in matching order. What follows is our evaluation results:

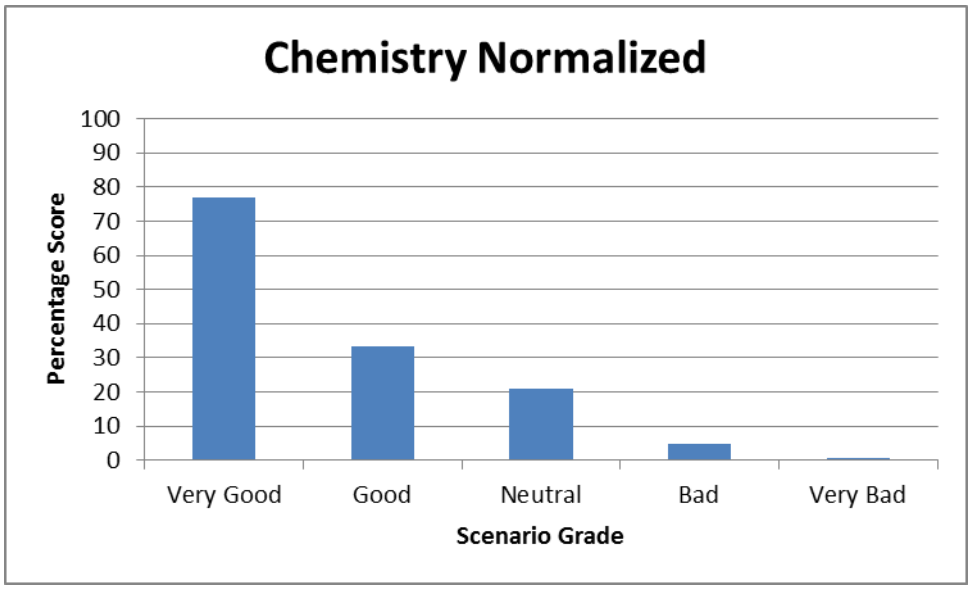

Figure 19: Evaluation method scores for chemistry scenario's maps

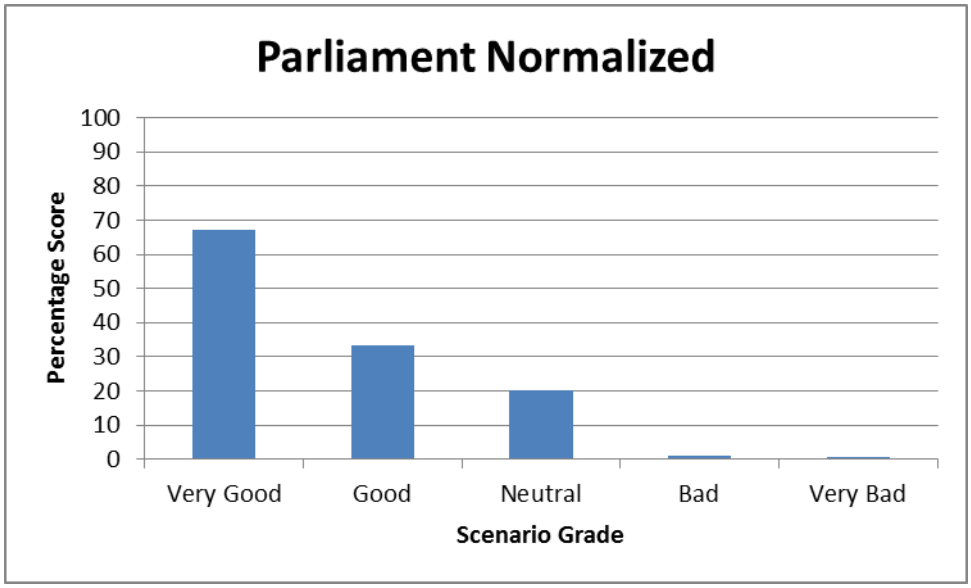

Figure 20: Evaluation method scores for parliament scenario's maps

Both of our scenarios have gratifyingly produced the expected result, with maps of each grade producing scores quite clearly differentiated from the grade above and below. The sharp drop in value between the neutral and bad maps particularly underlines how harshly our evaluation method punishes systems for badly-ranking an entity's relationships. This effect comes from relationships' dual roles of value-carriers and means of navigation, as a map is punished for poor ranking firstly when calculating the RLVs and then again when calculating each run's weight.

Our scores have given us the expected values for our scenarios, but we have yet to prove that the design of our scenarios will guarantee a corresponding user experience. It is at last time to check our scenarios against our user data and see if we can establish a broad correlation.

5.3.3 Relevant Relationships Marked by Users

Our first evaluation statistic is the number of relationships that users judged to be relevant that were also assessed as relevant beforehand by the scenario designer. This is a key measure for the performance of our evaluation method, as it would show that an expert assessor designing a scenario can stand in for dozens of users, accurately predicting which relationships users will recognize as useful and whether the tool in question will allow users to find them.

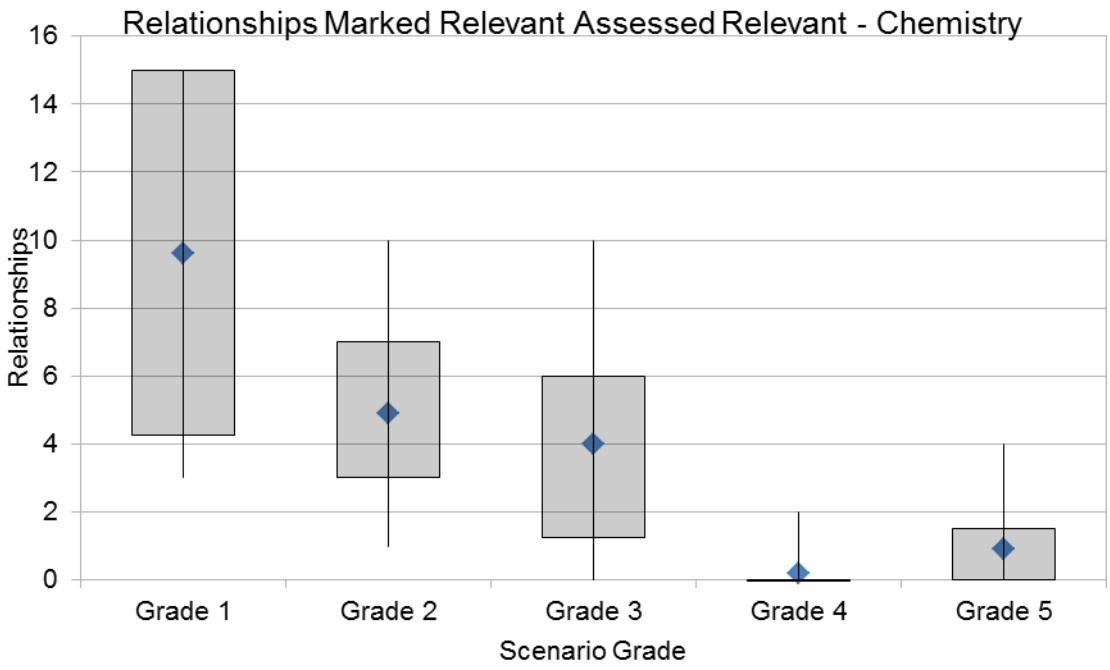

Figure 21: Number of relationships marked relevant by participants assessed relevant by assessor in the chemistry scenario

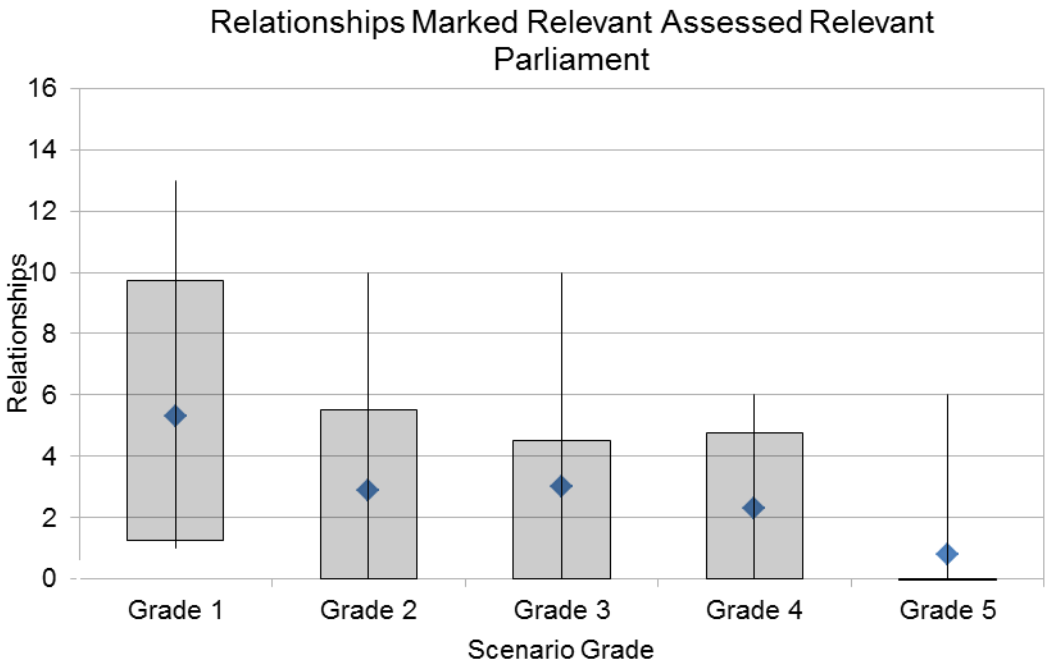

Figure 22: Number of relationships marked relevant by participants assessed relevant by assessor in the parliament scenario

As these figures clearly show, there is a direct correlation between a scenario's grade, its score by the evaluation method and the number of relationships assessed as relevant among the user's own chosen set. Similar results could be derived from figures 14 and 15 from the previous chapter, which showed the fraction of relationships marked relevant that were also assessed as relevant and which showed the same sort of trends. Again, the parliament scenario's weaker performance might be attributable to its ambiguity as an objective – it is a task less well-suited to entity-relationship recommenders, and that outcome is borne out here.

This is an excellent first step to validating our evaluation method, but it is not the whole picture. We might also want to establish that even if users fail to recognize relevance in their assessments, good maps will at least tend to lead them toward finding more relationships assessed by our assessor as relevant, hence our next data analysis.

5.3.4 Relevant Relationships Found by Users

This evaluative statistic represents the number of relationships among the thirty that users found over the course of the test that were assessed as relevant beforehand by the scenario designer. This statistic does not take into account whether users then judged them to be relevant as well, merely whether they added them to their subgraph.

The value of this statistic is to confirm that our template approach to designing the five grades of scenario maps had the desired effect of funneling users, either toward or away from assessed-relevant relationships depending on the grade. Finding a larger number of relevant relationships on high-value maps and failing to find them on low-value ones emphasizes the role that the quality of a system's output plays in directing a user's attention.

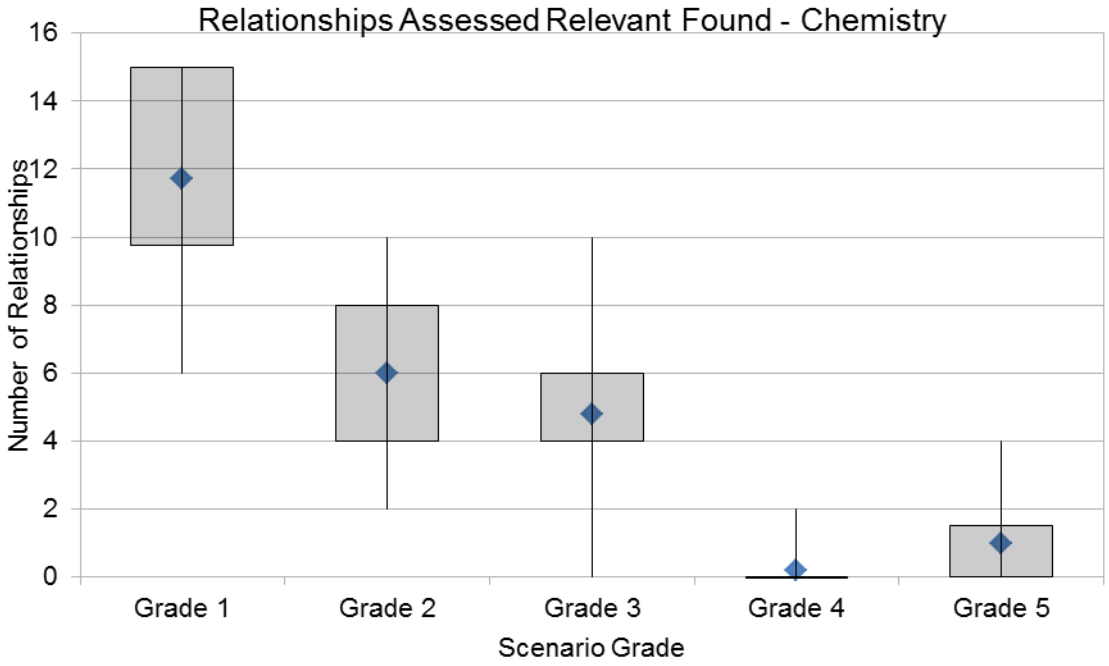

Figure 23: Relationships found by users completing the Chemistry scenario which were previously assessed as relevant

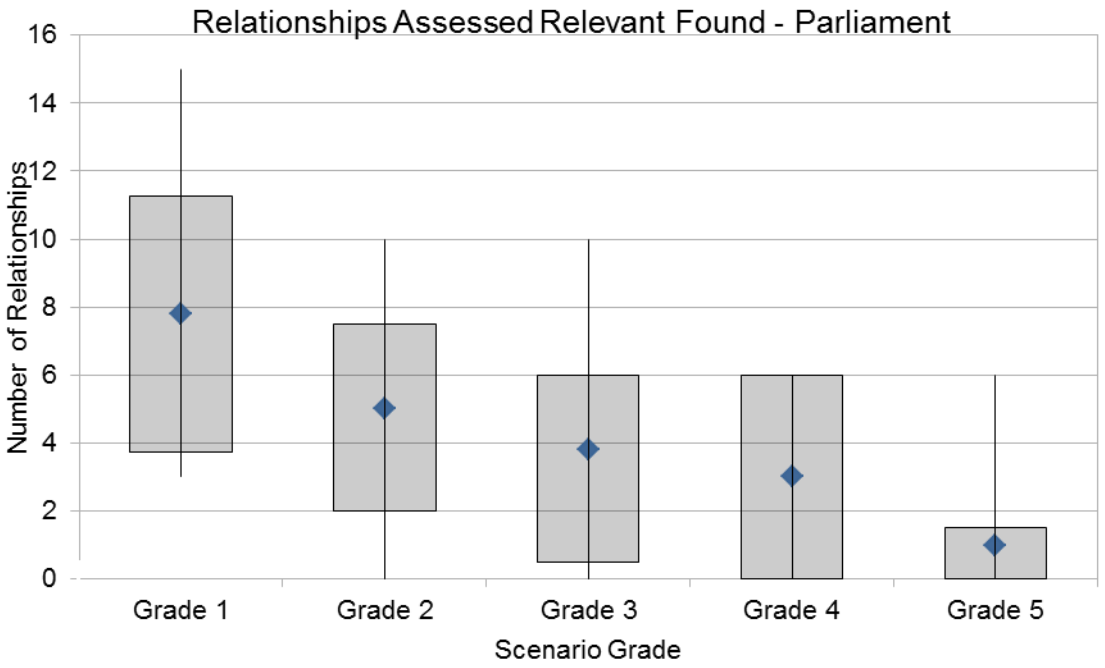

Figure 24: Relationships found by users completing the Parliament scenario which were previously assessed as relevant

Sure enough, these graphs display the trend clearly. The templates appear to have influenced how many relevant relationships users were presented with when asked to make their own judgments, vindicating our approach to producing the five grades of maps. This result undoubtedly helps explain the previous one, as even users marking relationships at random would be more likely to hit upon relevance on a high-quality map than a low one.

5.3.5 Conclusion

Taken together, our two user-data statistics validate our evaluation method by demonstrating a high degree of correlation. There is a definite relation between the scores devised by our method, the design of the templates producing each grade's map and the amount of relevant information (real and perceived) found by users.

The significance of this correlation for entity-relationship recommenders cannot be easily understated. Such a method is not merely a scoring script, it could represent the foundations for a

whole test track. This variety of system could be featured at a future TREC, spurring new research and allowing existing systems to compare their performance in a more neutral, public environment.

That does not guarantee that this method is perfect, or could not be improved. The degradation functions used for the RLVs and RWs are effectively arbitrary and could be fine-tuned to be more effective. We have also seen that this method's accuracy is tied to the ambiguity of the scenario, meaning tools meant to perform in ambiguous circumstances will be less surely evaluated. There is also an implicit assumption at work here that one scenario is much like another, ambiguity aside – an assumption that could possibly prove false, if someone could conceive of a reason to divide them.

These are but words of caution and considerations for future research, not existential threats to our method. For the moment, our work represents a step forward for entity-relationship systems as a field.

Chapter 6
Conclusion

And so we reach the end of our long, meandering journey. We set out with the goal to explore evaluation in information retrieval and advance it into new territory, as well as helping a new kind of information technology grow and mature. Along the way we have surveyed history and the field's present standing, conducted a sizeable study, tested behavioral theories and developed our own means of evaluating entity relationship recommender systems with an eye toward complex IR evaluation as a whole. How did we fare?

6.1 Recap

6.1.1 Summary of Contributions

On the behavior front, a lot of useful new data was gathered and analyzed for fresh insight. We have discovered that some traditional behavior models in information retrieval will still apply to new complex tasks, such as the importance of ranking to how people perceive relevance. Other models, like the theory of breadth vs. depth do not appear to apply. A lesson to be taken from this is we should not be too quick to embrace or dismiss existing IR wisdom when applying it to complex tasks, but rather to be thorough and conduct further tests.

We also gained a great deal of insight for future developers. The prevalence of depth-first search emphasizes that when users are building subgraphs the system cannot rely on them to go back the way they came and explore other avenues unless they hit a dead end. The natural tendency for users to lower their bar for relevance in the absence of any particularly useful information is also something to be mindful of, as presenting users with meager returns might only misinform them or lead them astray as they look for relevance that is not there. The importance of ranking has been reaffirmed in a new environment, both as a priority for systems looking to present available information in the most effective way possible and as a tool for evaluators when assessing system performance.

On evaluation, we have laid out the framework for a new method to gauge the performance of entity-relationship recommender systems. This method could one day drive a TREC track and spur the

development of the field, or at least spur evaluation within the field closer toward that goal. By blending elements of the traditional information retrieval paradigm with the more human-factor driven motivation of later, alternative paradigms our method enjoys the best of both worlds.

6.1.2 Flaws and Pitfalls

No research work is flawless, humility demands a moment to assess where we went wrong during the conduct of this research and mistakes that might be avoided in the future. For example, one definite limitation of our approach to building our method and study was the purely practical limits on scenario size and test scope. Constructing scenario maps manually was an arduous process, and some irregularities in the outcome might be attributed to hand-crafting imperfections. Future work in this field might seek to use entity and relationship extractors working on relevant documents to build entity-relationship graphs faster, assuming the issues of accuracy and cost can be overcome.

One issue encountered during the user study derived from the artificial nature of the objective given to users, the effect of which was observed during our discussion on ambiguity. The purpose of the study was to ascertain how users behave when pursuing a higher-level information need than a single, isolated query. As such, asking people to pursue an information need which is not their own is less effective and provides less real insight than if they were pursuing an actual, natural information need they developed on their own. The latter is difficult to arrange in a study, for obvious reasons, and reestablishes the need for an evaluation method that does not require engaging large crowds of test users, but its influence in the outcome of this study should be noted.

The study supporting much of this thesis is actually the third such study we have conducted in pursuit of this goal. The previous two studies, much more akin to pilots in scale and conception, gave several insights which proved critical in implementing the third. Some are more general observations about the nature of user studies, such as the importance of serious recruiting efforts and the technical challenges involved in producing and disseminating a test program.

Others are more relevant, such as the importance of implementing a proper latin square format for which maps users would complete. Due to the large number of cases involved in the study, following an improper format could have resulted in some cases having too few data points. The design of the

system's interface evolved between studies to become closer to what an actual entity-relationship recommender system might look like, a key point if we mean to argue that our results should be meaningful. Having the test program produce output pre-processed and formatted for analysis also saved a great deal of time compared to earlier, cruder implementations, the lesson here being to know what you intend to do with data before you start collecting it.

6.2 Future Work

6.2.1 Building Recommender Systems

Having a method of evaluation with many of the benefits of a Cranfield approach yet retaining a correlation to real user experience could be a major boon for the fledgling field of entity-relationship recommender systems. There are systems currently undergoing development, both within the research group that gave rise to this method and in the wider academic and private world, to which it might apply.

Having this means at their disposal is one of the ways a field grows and gains legitimacy. As the Cranfield paradigm itself leant the proceedings of the early Message Understanding Conferences an air of scientific authority, this method might similarly convince others of the legitimacy of entity-relationship recommenders. It is a first step toward recognition, something to point to when making the case that the subject is an active one drawing fresh interest.

Whether it will live up to this potential and find a place in academic usage or not is entirely another question, the answer to which cannot yet be known. Still, some optimism would not be unwarranted. At the very least, for a field still looking to grow, any activity at all is a good start.

6.2.2 The Future of Entity-Relationship Systems

Recommender systems are just one breed of entity-relationship system, and not even a particularly prolific one. There remains a great deal of work left to do in evaluation, but the principles and approaches outlined here may turn out to be adaptable to other systems.

For example, the centrality task that MING and CEPS both attempt - in that case it is the system, not the user, which builds a subgraph of relevant entities and relationships out of a larger whole. Could an

evaluation method for this be developed by taking our own approach, swapping the right roles and finding what statistic to use as the measure of performance? Certainly a method may want to achieve the same features as ours, such as creating reusable test collections, deriving hard numeric scores from the system's output and using user data to validate their results.

If one evaluation method can come from adapting another, there may be room for a general framework which includes both methods as variations of itself – and further methods for other kinds of systems. This goes beyond just entity-relationship systems, heading toward complex information retrieval as a whole.

6.2.3 Complex IR Evaluation

Entity-relationship systems are themselves just a small part of the complex information retrieval project. Larger frameworks already exist, covering advanced topics such as exploratory search and interactive information retrieval. Progress will continue to be made in these areas, drawing information retrieval away from its simple, familiar tools and into the future. As different kinds of systems overlap and merge, providing a wider range of support to users' information needs, their evaluation methods will likely follow suite.

This is somewhat of a grand subject for so humble a thesis as this, however. One more method for one kind of entity-relationship system is but a small brick in a large wall, itself part of soaring edifice. We can content ourselves knowing that our minor contribution exists, and that the larger project continues apace in some small part because of it.

6.3 Closing Remarks

It can be difficult to gauge the success of a tool before one has a chance to use it in earnest, an issue at the heart of the evaluation field. It has taken many pages and much time to say with some certainty that this is a worthy means to evaluate entity-relationship recommender systems, but it has at last been said. We are left one tool richer, and perhaps a little richer in our understanding of information retrieval and its history, for the journey. Certainly, the method will serve our own purposes, and research ongoing in our own small pond will benefit by it.

For a method such as ours, though, we cannot be the sole measure of success. Success for an evaluation method comes from being used to evaluate, not by its creators but by others persuaded of its sound judgments. Validation would not come from having the method published in a conference or journal – it comes from that paper being cited by others, who go looking for a way to evaluate their newly-conceived system and hit upon our new approach. It comes from new systems made specifically to take advantage of it or, perhaps the highest form of validation, a conference test track based on it.

Even if our method is not seized upon as a new standard, this would not make it a failure. It may be that disseminating our technique will lead to critique rather than adoption. This might lead to more researchers taking an interest in the question of evaluation, discussing where we have gone wrong and what needs to be done to better resolve the question of evaluation, and ultimately achieving our goal. Provoking this discussion is just as meaningful a way to advance the field.

Or, to reduce this thesis to a tired but familiar expression: if you build it, they will come.

Appendix A

User Study Data

Data type	File 1	File 2	File 3	File 4	File 5	File 6	File 7	File 8	File 9	File 10
Domain	4	2	2	3	3	3	4	4	3	3
Type A	0	0	3	1	1	0	0	0	2	0
Type B	4	3	0	3	2	4	3	4	1	4
Type C	0	1	1	0	1	0	1	0	1	0
Relationships Marked Relevant Actually Relevant	9	4	3	5	15	15	11	4	15	15
Relationships Marked Relevant Not Relevant	9	4	0	16	0	0	4	12	0	0
Relationships Not Marked But Actually Relevant	3	5	9	1	0	0	1	2	0	0
Total Relationships Marked	18	8	3	21	15	15	15	16	15	15
Total Relevant Relationships Among Entities Chosen By User	12	9	12	6	15	15	12	6	15	15
Total Relevant Relationships Across Whole Map	30	30	30	30	30	30	30	30	30	30
Choice 1 Rank	1	1	2	2	3	4	4	4	1	2
Choice 2 Rank	1	4	1	1	3	1	1	1	1	2
Choice 3 Rank	3	2	3	5	1	2	5	5	2	2
Choice 4 Rank	2	5	4	5	3	2	2	5	3	3
Choice 5 Rank	4	4	5	5	3	3	1	1	1	3
Average Choice Rank	2.2	3.2	3	3.6	2.6	2.4	2.6	3.2	1.6	2.4
# Added Entities Looked At	5	5	2	5	5	5	5	5	3	5

Table A1: Chemistry Grade 1 – Very Good results

Data Type	File 1	File 2	File 3	File 4	File 5	File 6	File 7	File 8	File 9	File 10
Domain	4	3	4	3	1	3	4	3	3	3
Type A	1	1	0	1	1	0	0	0	2	0
Type B	3	0	4	1	3	4	3	3	2	4
Type C	0	3	0	2	0	0	1	1	0	0
Relationships Marked Relevant Actually Relevant	3	2	4	10	4	10	4	1	8	3
Relationships Marked Relevant Not Relevant	12	16	3	0	12	0	18	7	0	10
Relationships Not Marked But Actually Relevant	3	0	4	0	0	0	0	3	0	1
Total Relationships Marked	15	18	7	10	16	10	22	8	8	13
Total Relevant Relationships Among Entities Chosen By User	6	2	8	10	4	10	4	4	8	4
Total Relevant Relationships Across Whole Map	20	20	20	20	20	20	20	20	20	20
Choice 1 Rank	5	5	4	3	4	2	3	2	2	4
Choice 2 Rank	2	1	1	1	5	1	5	5	1	1
Choice 3 Rank	4	5	3	3	3	3	4	5	3	4
Choice 4 Rank	3	4	3	3	5	1	5	3	2	4
Choice 5 Rank	1	4	4	1	5	1	1	2	3	2
Average Choice Rank	3	3.8	3	2.2	4.4	1.6	3.6	3.4	2.2	3
# Added Entities Looked At	5	4	5	4	5	5	4	4	4	5

Table A2: Chemistry Grade 2 – Good results

Data Type	File 1	File 2	File 3	File 4	File 5	File 6	File 7	File 8	File 9	File 10
Domain	3	3	3	3	4	4	4	3	3	4
Type A	2	0	2	0	0	1	4	0	1	1
Type B	1	3	1	4	4	2	0	4	3	3
Type C	1	1	1	0	0	1	0	0	0	0
Relationships Marked Relevant Actually Relevant	0	6	10	6	2	2	1	4	1	8
Relationships Marked Relevant Not Relevant	13	1	0	3	9	25	13	1	12	5
Relationships Not Marked But Actually Relevant	0	0	0	0	2	2	3	0	1	0
Total Relationships Marked	13	7	10	9	11	27	14	5	13	13
Total Relevant Relationships Among Entities Chosen By User	0	6	10	6	4	4	4	4	2	8
Total Relevant Relationships Across Whole Map	20	20	20	20	20	20	20	20	20	20
Choice 1 Rank	1	2	2	1	1	2	5	5	1	1
Choice 2 Rank	3	1	2	2	4	4	2	2	2	2
Choice 3 Rank	1	3	5	5	2	3	1	4	5	2
Choice 4 Rank	5	2	2	5	3	5	4	5	3	2
Choice 5 Rank	3	2	5	4	5	4	3	3	1	2
Average Choice Rank	2.6	2	3.2	3.4	3	3.6	3	3.8	2.4	1.8
# Added Entities Looked At	3	5	3	5	5	3	1	5	5	5

Table A3: Chemistry Grade 3 – Neutral results

Data Type	File 1	File 2	File 3	File 4	File 5	File 6	File 7	File 8	File 9	File 10
Domain	3	3	3	3	2	3	3	1	4	4
Type A	1	0	1	0	0	0	1	2	1	2
Type B	2	4	1	4	2	4	2	1	1	1
Type C	1	0	2	0	2	0	1	1	2	1
Relationships Marked Relevant Actually Relevant	0	0	0	0	0	0	0	2	0	0
Relationships Marked Relevant Not Relevant	6	7	13	6	3	14	0	5	8	14
Relationships Not Marked But Actually Relevant	0	0	0	0	0	0	0	0	0	0
Total Relationships Marked	6	7	13	6	3	14	0	7	8	14
Total Relevant Relationships Among Entities Chosen By User	0	0	0	0	0	0	0	2	0	0
Total Relevant Relationships Across Whole Map	20	20	20	20	20	20	20	20	20	20
Choice 1 Rank	3	1	1	5	4	2	2	2	1	1
Choice 2 Rank	4	1	4	3	4	2	3	3	2	2
Choice 3 Rank	3	3	3	1	2	4	4	1	2	2
Choice 4 Rank	5	3	1	2	2	5	3	5	3	4
Choice 5 Rank	2	1	4	4	3	3	1	5	5	5
Average Choice Rank	3.4	1.8	2.6	3	3	3.2	2.6	3.2	2.6	2.8
# Added Entities Looked At	5	5	5	5	5	5	5	3	3	5

Table A4: Chemistry Grade 4 – Bad results

Data Type	File 1	File 2	File 3	File 4	File 5	File 6	File 7	File 8	File 9	File 10
Domain	4	3	2	3	3	4	2	3	3	2
Type A	2	0	0	1	0	0	2	2	0	2
Type B	2	4	3	2	3	4	1	1	4	2
Type C	0	0	1	1	1	0	1	1	0	0
Relationships Marked Relevant Actually Relevant	2	3	4	0	0	0	0	0	0	0
Relationships Marked Relevant Not Relevant	22	11	0	19	22	24	7	5	23	15
Relationships Not Marked But Actually Relevant	0	1	0	0	0	0	0	0	0	0
Total Relationships Marked	24	14	4	19	22	24	7	5	23	15
Total Relevant Relationships Among Entities Chosen By User	2	4	4	0	0	0	0	0	0	0
Total Relevant Relationships Across Whole Map	20	20	20	20	20	20	20	20	20	20
Choice 1 Rank	5	3	1	4	4	2	5	1	1	1
Choice 2 Rank	4	5	4	5	2	5	1	1	3	2
Choice 3 Rank	2	5	5	4	3	2	4	4	4	1
Choice 4 Rank	3	4	5	2	1	1	5	5	3	4
Choice 5 Rank	5	3	5	3	4	5	1	4	5	5
Average Choice Rank	3.8	4	4	3.6	2.8	3	3.2	3	3.2	2.6
# Added Entities Looked At	3	5	4	5	5	5	5	3	5	4

Table A5: Chemistry Grade 5 – Very Bad results

Data Type	File 1	File 2	File 3	File 4	File 5	File 6	File 7	File 8	File 9	File 10
Domain	2	2	2	2	5	2	3	2	1	3
Type A	0	0	0	0	3	0	0	1	0	1
Type B	4	4	3	4	0	3	4	2	3	2
Type C	0	0	1	0	1	1	0	1	1	1
Relationships Marked Relevant Actually Relevant	1	1	2	13	11	12	6	4	2	1
Relationships Marked Relevant Not Relevant	11	16	12	0	18	0	0	10	3	12
Relationships Not Marked But Actually Relevant	2	2	4	2	1	0	0	5	7	2
Total Relationships Marked	12	17	14	13	29	12	6	14	5	13
Total Relevant Relationships Among Entities Chosen By User	3	3	6	15	12	12	6	9	9	3
Total Relevant Relationships Across Whole Map	30	30	30	30	30	30	30	30	30	30
Choice 1 Rank	1	1	1	1	1	1	1	1	1	1
Choice 2 Rank	4	5	4	1	2	4	4	5	3	4
Choice 3 Rank	5	5	3	2	3	2	5	3	5	5
Choice 4 Rank	4	4	4	2	5	3	3	5	5	4
Choice 5 Rank	1	1	4	2	4	3	3	5	5	4
Average Choice Rank	3	3.2	3.2	1.6	3	2.6	3.2	3.8	3.8	3.6
# Added Entities Looked At	5	5	5	5	2	4	5	4	5	5

Table A6: Parliament Grade 1 – Very Good results

Data Type	File 1	File 2	File 3	File 4	File 5	File 6	File 7	File 8	File 9	File 10
Domain	1	2	1	3	2	2	1	2	1	4
Type A	0	1	1	1	0	0	0	4	2	1
Type B	3	1	3	3	4	4	3	0	2	2
Type C	1	2	0	0	0	0	1	0	0	1
Relationships Marked Relevant Actually Relevant	0	0	4	10	0	0	8	0	6	1
Relationships Marked Relevant Not Relevant	7	19	4	0	14	16	10	5	1	9
Relationships Not Marked But Actually Relevant	2	0	4	0	2	2	0	6	0	5
Total Relationships Marked	7	19	8	10	14	16	18	5	7	10
Total Relevant Relationships Among Entities Chosen By User	2	0	8	10	2	2	8	6	6	6
Total Relevant Relationships Across Whole Map	20	20	20	20	20	20	20	20	20	20
Choice 1 Rank	1	1	2	3	1	1	1	2	1	3
Choice 2 Rank	1	2	1	1	5	1	4	3	4	5
Choice 3 Rank	5	4	4	1	1	2	3	4	3	2
Choice 4 Rank	5	5	3	3	5	4	5	1	1	1
Choice 5 Rank	4	4	3	3	5	3	5	5	3	1
Average Choice Rank	3.2	3.2	2.6	2.2	3.4	2.2	3.6	3	2.4	2.4
# Added Entities Looked At	5	5	5	5	5	5	4	2	5	4

Table A7: Parliament Grade 2 – Good results

Data Type	File 1	File 2	File 3	File 4	File 5	File 6	File 7	File 8	File 9	File 10
Domain	1	2	4	3	3	3	2	1	2	1
Type A	3	0	2	2	2	0	0	0	0	1
Type B	0	4	2	0	1	4	3	4	3	2
Type C	1	0	0	2	1	0	1	0	1	1
Relationships Marked Relevant Actually Relevant	3	0	10	0	10	5	0	0	2	0
Relationships Marked Relevant Not Relevant	1	17	0	16	0	16	9	1	12	5
Relationships Not Marked But Actually Relevant	3	2	0	0	0	1	2	0	0	0
Total Relationships Marked	4	17	10	16	10	21	9	1	14	5
Total Relevant Relationships Among Entities Chosen By User	6	2	10	0	10	6	2	0	2	0
Total Relevant Relationships Across Whole Map	20	20	20	20	20	20	20	20	20	20
Choice 1 Rank	1	1	2	1	5	2	2	3	2	1
Choice 2 Rank	3	2	5	4	2	2	4	3	4	4
Choice 3 Rank	2	1	2	1	5	4	3	4	3	1
Choice 4 Rank	5	1	5	3	5	1	5	5	3	3
Choice 5 Rank	5	5	2	3	2	4	3	3	3	4
Average Choice Rank	3.2	2	3.2	2.4	3.8	2.6	3.4	3.6	3	2.6
# Added Entities Looked At	2	5	5	3	5	5	5	5	5	5

Table A8: Parliament Grade 3 – Neutral results

Data Type	File 1	File 2	File 3	File 4	File 5	File 6	File 7	File 8	File 9	File 10
Domain	2	2	1	3	1	3	1	2	2	1
Type A	2	0	0	0	1	1	1	1	1	0
Type B	2	3	4	4	2	2	1	3	2	4
Type C	0	1	0	0	1	1	2	0	1	0
Relationships Marked Relevant Actually Relevant	0	4	0	1	6	6	0	5	1	0
Relationships Marked Relevant Not Relevant	20	2	7	5	9	12	16	16	11	20
Relationships Not Marked But Actually Relevant	0	0	0	5	0	0	0	1	1	0
Total Relationships Marked	20	6	7	6	15	18	16	21	12	20
Total Relevant Relationships Among Entities Chosen By User	0	4	0	6	6	6	0	6	2	0
Total Relevant Relationships Across Whole Map	20	20	20	20	20	20	20	20	20	20
Choice 1 Rank	4	1	1	1	1	1	1	1	1	1
Choice 2 Rank	2	2	1	4	2	4	2	2	5	3
Choice 3 Rank	4	5	4	3	4	2	3	5	2	1
Choice 4 Rank	1	1	4	2	5	5	1	4	4	4
Choice 5 Rank	2	5	4	5	3	3	1	3	1	5
Average Choice Rank	2.6	2.8	2.8	3	3	3	1.6	3	2.6	2.8
# Added Entities Looked At	3	5	5	5	5	5	5	5	5	5

Table A9: Parliament Grade 4 – Bad results

Data Type	File 1	File 2	File 3	File 4	File 5	File 6	File 7	File 8	File 9	File 10
Domain	2	1	3	1	2	2	2	3	3	1
Type A	0	0	0	0	0	0	0	2	2	1
Type B	4	4	4	3	2	4	4	2	1	2
Type C	0	0	0	1	2	0	0	0	1	1
Relationships Marked Relevant Actually Relevant	6	0	0	0	0	0	0	2	0	0
Relationships Marked Relevant Not Relevant	0	8	14	5	16	7	17	12	9	14
Relationships Not Marked But Actually Relevant	0	0	2	0	0	0	0	0	0	0
Total Relationships Marked	6	8	14	5	16	7	17	14	9	14
Total Relevant Relationships Among Entities Chosen By User	6	0	2	0	0	0	0	2	0	0
Total Relevant Relationships Across Whole Map	20	20	20	20	20	20	20	20	20	20
Choice 1 Rank	4	1	4	1	3	1	5	3	1	1
Choice 2 Rank	5	1	1	1	5	1	2	5	5	4
Choice 3 Rank	5	4	4	3	5	2	1	1	4	4
Choice 4 Rank	4	3	5	1	1	4	5	4	2	2
Choice 5 Rank	5	2	3	3	3	2	1	5	5	2
Average Choice Rank	4.6	2.2	3.4	1.8	3.4	2	2.8	3.6	3.4	2.6
# Added Entities Looked At	5	5	5	5	5	5	5	5	5	5

Table A10: Parliament Grade 5 – Very Bad results

Appendix B

Glossary

Entity: An individual and discrete object, place, person, event, thing or so on.

Relationship: What tie together separate **Entities**. While relationships can theoretically be complex enough to tie together multiple entities at once, for simplicity's sake relationships are usually imagined as connecting only two entities each.

Entity-Relationship Model: An organizational theory for information, where things such as events, people, places and so on are distilled into **Entities** and their connections and interactions are recorded as **Relationships**. Often this model is realized on a **Graph**.

Entity-Relationship System: An information retrieval implementation of the **Entity-Relationship Model**.

Entity Relationship Recommender System: A subtype of **Entity-Relationship Systems**. Recommender systems are distinguished by having the users build their own **Graph** of **Entities** from a larger one by recommending **Relationships** it thinks is relevant to a user's exploration goals.

Exploratory Search: A modern alternative paradigm in **Information Retrieval**, supporting users' more complex **Information Needs** by enabling the exploration of a larger knowledge space.

Information Retrieval (IR): A research field within computer science dedicated to providing users with timely and useful information.

Cranfield Paradigm: An evaluation paradigm within **Information Retrieval** developed from the work of Cyril Cleverdon at Cranfield University.

Precision: The percentage of retrieved information classified as relevant.

Recall: The percentage of available relevant information successfully retrieved.

F1-Score: The harmonic mean of **Precision** and **Recall**. Often referred to as accuracy, one of the main means of measuring success in the **Cranfield Paradigm**.

Human-Computer Interaction (HCI): A research field within computer science dedicated to

Domain: A semantic subject of interest.

Discounted Cumulative Gain: A formula used to calculate the total value of a sequence of values, which each value weighted according to its place in the sequence. Weights decrease the further along the sequence each value is.

Relevance: Whether information is thought to be useful in fulfilling a user's **Information Need**. Exact definition of what makes information relevant or not a subject of debate.

Information Need: A desire by a user to learn something which has driven them to interact with **Information Retrieval** systems. Normally imagined as a short, direct and clearly-formulated query by traditional IR, complex IR tasks have begun to tackle larger and more vague needs made up of many smaller needs – even needs which the user only formulates over the course of interacting with the system.

Graph: A popular form of data organization and conceptualization, with accompanying algorithmic principles. A graph is typically made up of a network of **Nodes** connected by **Edges**. Implementations of the **Entity-Relationship Model** are often illustrated as graphs.

Node: A single, discrete point on a **Graph**. Corresponds to an **Entity** in the **Entity-Relationship Model**.

Edge: A connection between two **Nodes** on a **Graph**. Corresponds to a **Relationship** in the **Entity-Relationship Model**.

Bibliography

Azzopardi, L., & Vishwa, V. (2008). Retrievability: an evaluation measure for higher order information access tasks. *Proceedings of the 17th ACM conference on Information and knowledge management* (pp. 561-570). New York: ACM.

Balog, K., De Vries, A. P., & Serdyukov, P. (2011). Overview of the TREC 2011 Entity Track. *Proceedings of the Twentieth Text REtrieval Conference (TREC 2011).* NIST.

Blanco, R., Halpin, H., Herzig, D. M., Mika, P., Pound, J., & Thompson, H. S. (2011). Entity Search Evaluation over Structured Web Data. *EOS 2011 - Entity Oriented Search Workshop.* Beijing.

Borlund, P. (2003). The Concept of Relevance in IR. *Journal of the American Society for Information Science and Technology*, 913-925.

Chen, P. (1976). The entity-relationship model—toward a unified view of data. *ACM Transactions on Database Systems* (pp. 9-36). New York: ACM.

Chinchor, N., & Sundheim, B. (1993). MUC-5 evaluation metrics. *Proceeding MUC5 '93 Proceedings of the 5th conference on Message understanding* (pp. 69-78). Stroudsburg: Association for Computational Linguistics.

Clarke, C. L., Kolla, M., Gordon, C. V., Vechtomova, O., Ashkan, A., Buttcher, S., et al. (2008). Novelty and diversity in information retrieval evaluation. *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 659-666). New York: ACM.

Geng, L., & Hamilton, H. J. (2006). Interestingness measures for data mining: A survey. *ACM Computing Surveys*.

Joachims, T. (2002). Optimizing search engines using clickthrough data. *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 133-142). New York: ACM.

Kasneci, G., Elbassuoni, S., & Weikum, G. (2009). MING: mining informative entity relationship subgraphs. *Proceedings of the 18th ACM conference on Information and knowledge management* (pp. 1653-1656). New York: ACM.

Nitsche, M., & Nurnberger, A. (2012). Trailblazer - Towards the Design of an Exploratory Search User Interface. *Proceedings of HCIR 2012.* Cambridge.

Novak, J. (1977). *A Theory of Education.* Cornell University Press.

Russell-Rose, T., & Makri, S. (2012). Designing for Consumer Search Behaviour. *Proceedings of HCIR 2012.* Cambridge.

Russell-Rose, T., & Tate, T. (2012). *Designing the Search Experience: The Information Architecture of Discovery*. Morgan Kaufmann.

Smucker, M., & Clarke, C. (2012). The Fault, Dear Researchers, is not in Cranfield, But in our Metrics, that they are Unrealistic. *EuroHCIR2012*, (pp. 11-12).

Tong, H., & Faloutsos, C. (2006). Center-piece subgraphs: problem definition and fast solutions. *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 404-413). New York: ACM.

Voorhees, E. (2001). The Philosophy of Information Retrieval Evaluation. *Proceedings of the The Second Workshop of the Cross-Language Evaluation Forum on Evaluation of Cross-Language Information Retrieval Systems* (pp. 355-370). London: Springer-Verlag.

Voorhees, E. M., & Harman, D. K. (2005). *TREC: Experiment and Evaluation in Information Retrieval.* Cambridge: MIT Press.

White, R. W., Marchionini, G., & Muresan, G. (2008). Evaluating Exploratory Search Systems. *Information Processing and Management*.

White, R., & Roth, R. A. (2009). *Exploratory Search Beyond the Query–Response Paradigm.* Morgan & Claypool.

Wildemuth, B. M., & Freund, L. (2012). Assigning search tasks designed to elicit exploratory search behaviors. *Proceedings of the Symposium on Human-Computer Interaction and Information Retrieval* . New York: ACM.

Yogev, S., Roitman, H., Carmel, D., & Zwerdling, N. (2012). Towards Expressive Exploratory Search Over Entity-Relationship Data. *roceedings of the 21st international conference companion on World Wide Web* (pp. 83-92). New York: ACM.

Zarro, M. (2012). Developing A Dual-Process Information Seeking Model for Exploratory Search. *Proceedings of HCIR 2012.* Cambridge.

Zhang, X., Gao, Z., & Gui, Y. (2013). A Comparative Study on Statistical Classifcation Methods in Relation Extraction. *Proceedings of the 3rd International Conference on Multimedia Technology.* Guangzhou.