Probabilistic, Object-oriented Logics for Annotation-based Retrieval in Digital Libraries

Ingo Frommholz
ingo.frommholz@uni-due.de

Norbert Fuhr
fuhr@uni-duisburg.de

University of Duisburg-Essen
D-47048 Duisburg
Germany

ABSTRACT

In this paper we introduce POLAR, a probabilistic object-oriented logical framework for annotation-based information retrieval. In POLAR, the knowledge about digital objects, annotations and their relationships in a digital library repository can be modelled considering certain characteristics of annotations and annotated objects. Insights about these characteristics are gained by an analysis of the annotation models behind existing systems and a discussion of an object-oriented, logical view on relevant objects in a digital library. Retrieval methods applied in a digital library should take annotations into account to satisfy users' information needs. POLAR thus supports a wide range of flexible and powerful annotation-based fact and content queries by making use of knowledge and relevance augmentation. An evaluation of our approach on email discussions shows performance improvements when annotation characteristics are considered.

Categories and Subject Descriptors: H.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing; H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval

General Terms: Algorithms, Design

Keywords: Annotations, probabilistic Datalog, annotation-based retrieval, POOL, POLAR

1. INTRODUCTION

Several tasks of digital libraries like the creation, management, retrieval and effective use of documents can be supported by annotations [3]. Such annotations manifest themselves in different forms and dimensions, ranging from simple highlighting of text and personal notes through (typed) links between documents up to nested and shared annotations with which collaborative discussions about a specific topic are realised [18]. Besides in digital library systems like COLLATE [24] and DAFFODIL [17], annotation-based discussions can also be found in online newswire systems like ZDNet News[1], online encyclopedias like Wikipedia[2] and even email discussion lists.

If an object in a digital library is annotated, each annotation establishes a certain kind of document context, the context of "what others said about a document or its content". This context contains useful information exploitable for information retrieval in several ways. The main aim of this paper is to discuss possible annotation-based indexing and retrieval options by presenting POLAR, a probabilistic, object-oriented logical representation of documents and annotations with which a broad variety of retrieval strategies can be realised. First we will discuss a logical, object-oriented view on annotations and other relevant objects on the class and instance level in digital libraries. The purpose of this view is to gain an understanding of the aspects of annotations and how they relate to other digital objects managed in the repository. Based on this logical view, we are going to discuss POLAR, which can be used for indexing and representing our knowledge base and perform annotation-based document and discussion search by posing complex queries to the underlying knowledge base. Preliminary experiments with email discussions, which are presented afterwards, show that our model is suitable for annotation-based information retrieval.

2. DIGITAL LIBRARY OBJECTS

Our considerations are based on the entity-relationship model presented in [3], but in contrast to this model, we present an object-oriented view on the digital objects and annotations. In this view, which is depicted in Figure 1, we do not care where digital objects and annotations are actually stored, nor which scope (private, shared or public) they have; we assume that a service responsible for indexing and retrieval of digital objects and annotations has access to the required resources the user is allowed to retrieve. To describe objects and their relations, we sometimes use the syntax known from Description Logics [6].

JCDL'06, June 11–15, 2006, Chapel Hill, North Carolina, USA.

[1] http://news.zdnet.com/
[2] http://www.wikipedia.org/

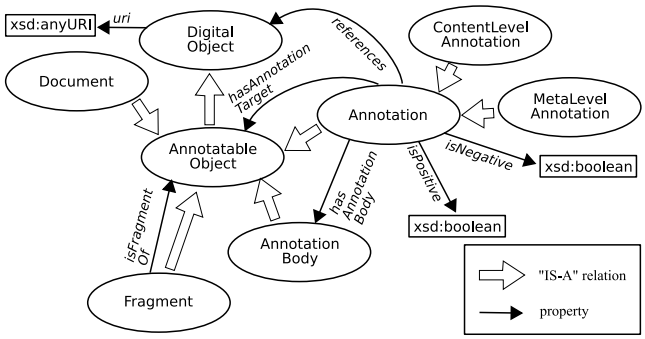

Figure 1: Classes and properties of the object-oriented view

2.1 Main Classes

2.1.1 Annotatable Objects and Annotations

A digital library manages digital objects of various kinds. We therefore need a class DigitalObject. Digital objects are identified by exactly one Uniform Resource Identifier (URI). Such digital objects might be, e.g., textual and multimedia documents, but also other things like digital representations of persons and conferences. In fact, a digital library might be able to handle a heterogeneous set of (possibly structured) digital objects. An *annotatable object* is therefore a digital object which can be annotated. We define a new class of annotatable objects as a subclass of digital objects:

$$\mathsf{AnnotatableObject} \sqsubseteq \mathsf{DigitalObject}$$

Let us assume a digital library where documents are annotatable. We identify a new class $\mathsf{Document} \sqsubseteq \mathsf{AnnotatableObject}$. Note that not only documents might be annotatable in a digital library. For example, in the DAFFODIL system [17], each user has a personal library consisting of certain user-defined folders which can also be annotated (for instance to give a short description).

So far we discussed annotatable objects without introducing annotations themselves. Annotations can be seen as digital objects as well[3], so we introduce a class $\mathsf{Annotation} \sqsubseteq \mathsf{DigitalObject}$. As explained in [3], annotations are metadata, strongly connected to the object they refer to. This means that each annotation must have at least one specific *annotation target*, which can be any annotatable object in the digital library repository. This is expressed by the property hasAnnotationTarget. When annotations can be nested (annotated again), we have $\mathsf{Annotation} \sqsubseteq \mathsf{AnnotatableObject}$. Annotations might simply be links connecting an annotatable object with another digital object. With the references property we are able to establish multiple links between digital objects: each annotation target is a source of the link, whereas each referenced object is its destination. Annotations can therefore be a means to connect certain objects to find hidden relationships or to establish new exploration paths within the documents in a digital library [4].

We did not talk about the content of annotations so far. Such content might consist of plain text, graphical symbols, structured text, and even multimedia content (like in the MADCOW system [7]). What exactly the "content" of annotations is should be subject to the actual application. If annotations are annotatable again, the question raises if an annotation itself or only its content is subject to annotation. We think that both annotations and their content should be annotatable. If an annotation has some text content and also references another digital object with the references property, another annotation might only refer to the reference, only to the text content, or to both, respectively. For this reason, and for the reason that the annotation content might be very complex, we introduce a new class $\mathsf{AnnotationBody} \sqsubseteq \mathsf{DigitalObject}$ conveying the actual content of an annotation. An instance of AnnotationBody must be the body of exactly one annotation. $\mathsf{AnnotationBody} \sqsubseteq \mathsf{AnnotatableObject}$ holds if annotations are annotatable again in the system.

The model presented so far still has some shortcomings; neither is it possible to annotate parts of documents (document fragments), nor can we define annotation types. These enhancements of the basic model will be introduced in the next two subsections.

2.1.2 Fragments

Fragments are certain portions of digital objects, in the case of an annotation scenario these are areas selected by the user as an annotation target. Such a fragment can be, for instance, a passage or paragraph of a text document, a certain video sequence, an excerpt of an image or a certain node (including its subtree) in an XML document. The Web annotation tool Annotea [16] uses XPointer[4] to mark a range of an XML (or HTML) document as annotation target. The Multivalent Browser[5] [22] provides means for annotations within different document formats; here, it is possible to mark a span and attach an annotation to this passage. The COLLATE prototype [24] allows for annotating multipaged scanned documents; users can annotate the whole document, a single page or a part of a page by marking an area of the scanned image and write a comment on it. Overall, the ability to identify a fragment of a document and to annotate it is a crucial functionality in a digital library supporting annotations. To reflect this in our model, we introduce a new class $\mathsf{Fragment} \sqsubseteq \mathsf{AnnotatableObject}$. Since fragments are created during the annotation process, they are related to the object they are part of. To connect fragments and their source object, we introduce a new property isFragmentOf and say that a fragment belongs to exactly one annotatable object. Due to the fact that the annotatable objects in a digital library might be of different kinds (textual, multimedia), it might be necessary to define subclasses of Fragment which adhere to the special characteristics of the fragmentable object. For example, in videos, fragments might be still images, whereas in text documents, fragments could be quotations.

2.1.3 Annotation Types

Some annotation systems offer a categorisation of annotations into several types. For example, in the COLLATE annotation model [24], discourse structure relations (like 'interpretation', 'support argument' and 'counterargument') are

[3]At least from a logical point of view. From a physical point of view, annotations can be managed by and stored in a separate annotation system independent from the digital library repository.

[4]`http://www.w3.org/TR/xptr-framework/`

[5]`http://multivalent.sourceforge.net/`

used as annotation types to allow for scientific discussion with nested annotations. Annotea knows several annotation types (or annotation classes) like 'comment', 'example', 'question' and 'explanation' [16], and also MADCOW allows for selecting certain annotation types [7]. According to the considerations in [3], annotations can contain *additional content* or *meta content*. Comments, for example, contain additional content and expand the content of the object they refer to. On the other hand, highlightings representing highlight markings operate on the meta level; if a passage is highlighted, the implicit assertion is "this part is important", but there is no additional content. Another example of meta level content are judgements, where people state their opinion about a document. To distinguish between these kinds of annotation types, we create new classes ContentLevelAnnotation $\sqsubseteq$ Annotation and MetaLevelAnnotation $\sqsubseteq$ Annotation and categorise our example annotation types accordingly:

$$\begin{aligned} \text{Highlighting} &\sqsubseteq \text{MetaLevelAnnotation} \\ \text{Judgement} &\sqsubseteq \text{MetaLevelAnnotation} \\ \text{Comment} &\sqsubseteq \text{ContentLevelAnnotation} \end{aligned}$$

2.1.4 Polarity

Another attribute of annotations is their *polarity* (if known). As an example, annotation types like "agreement" and "support argument" have a clear positive polarity, since they express a positive sentiment about the annotated part. In contrast, "disagreement" and "counterargument" convey a negative sentiment about the annotated content. In these cases, the annotation type determines the overall polarity of the annotation. In other cases, the polarity might not be clearly derivable from the annotation type, so the polarity might be determined by the annotation content (for example, a comment itself does not have a certain polarity, but there can of course be negative or positive statements in the comment). For Annotation, we define new functional properties isPositive and isNegative with an `xsd:boolean` datatype. It should be avoided to have both properties set to *true* since in this case we would have inconsistent knowledge.

2.2 Annotation Hypertext

In the last subsection we described the classes contained in our logical view. On the instance level we can identify another important structure: annotation hypertexts. We describe the definition of an annotation hypertext (or document-annotation hypertext) which is similar to the one [2]. $C(o)$ means that an instance o belongs to a class C, while $R(a, b)$ means instance a has value b for the property R.

Definition 1. *An* annotation hypertext *is a directed acyclic graph $G = (N, E)$ with N as the set of nodes and $o \in N$ iff* DigitalObject(o). *$E \subseteq N \times N$ is the set of edges with $(n, m) \in E$ iff* hasAnnotationTarget(n, m) *or* references(n, m) *or* isFragmentOf(n, m)*; other properties are not considered in the annotation hypertext.*

Taking into account the discussion in [2], defining an annotation hypertext as acyclic is reasonable. From a temporal point of view, fragments can only relate to digital objects that are older than them, and annotations can only annotate or reference older digital objects. This means that each object on a path from an annotation or fragment to a document leaf is older than its predecessor. If we would allow cycles in the annotation thread, this would not be the case any more; to close a cycle, an older object must annotate, reference or be a fragment of a younger one or itself, which is not possible.

Figure 2 shows an example of an annotation hypertext. a1 annotates d1 and a fragment of d2, f1_d2. a2 annotates a fragment of a1, f1_a1, and references a3.

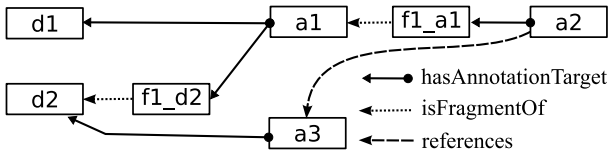

Figure 2: Example annotation hypertext

From an access-centric point of view, it is reasonable to reverse the directions of the edges in the annotation hypertext, reflecting more the direction in which users access objects in the hypertext. In the example in Fig. 2,users might access d2 first, move on to and read f1_d2, read a1, move on f1_a1 and finally read a2.

3. THE POLAR FRAMEWORK

As described by van Rijsbergen, information retrieval can be seen as uncertain inference [25]. We follow this idea and present a probabilistic, object-oriented representation of (structured) documents and annotation hypertexts. Our representation framework is called POLAR (Probabilistic Object-Oriented Logics for Annotation-based Retrieval). Within POLAR it is possible to pose queries to the knowledge base for retrieving documents (with the help of annotations) or relevant annotations. POLAR is similar to and very much influenced by POOL, another object-oriented, probabilistic logical model used for representing structured documents [23, 12]. POOL copes with structures which are trees from an access-centric point of view, whereas annotation hypertexts are not necessarily trees in this case; furthermore, POOL neither supports fragments nor different kinds of annotations. Thus, we have to extend POOL in order to represent such structures, especially for information retrieval. Like POOL, POLAR deals with *propositions* which can be *terms*, *classifications* or *properties*. Each proposition can be assigned a *probability*; if none is given, a probability of 1 is assumed. Another important elements of both POOL and POLAR are *rules* which consist of a head and a body.

3.1 Expressing the Knowledge Base in POLAR

3.1.1 Classes and Is-A Relations

In this subsection, we are going to discuss the representation of the object-oriented model introduced in Section 2.1. Representations of classes and Is-A relations in POLAR are the same as in POOL. The rules

```
metaLevelAnnotation(O) :- highlighting(O)
metaLevelAnnotation(O) :- judgement(O)
contentLevelAnnotation(O) :- comment(O)
annotation(O) :- metaLevelAnnotation(O)
```

```
annotation(O) :- contentLevelAnnotation(O)
annotatableObject(O) :- annotation(O)
annotatableObject(O) :- document(O)
digitalObject(O) :- annotatableObject(O)
```

express the "Is-A" relations in our model (capital letters denote variables); every instance which is a highlighting or judgement is also a meta level annotation, every comment is a content level annotation, and so on. Note that we did not list the AnnotationBody and Fragment classes here; these play a special role in POLAR as we will see later. With categorisations, we assign instances to their corresponding classes; for example,

```
document(d1)
annotation(a1)
```

means document($d1$) and annotation($a1$), respectively. Due to the above rules, `annotatableObject(d1)` and `annotatableObject(a1)` are intensional knowledge derived implicitly.

3.1.2 Document and Annotation Content and Structure

Documents and annotations in POLAR are represented by probabilistic propositions. These propositions are derived from the document and annotation content and annotation hypertext in an indexing process. Textual content is the source for term propositions in POLAR; their probability can be estimated using traditional tf-based measures normalised to the range between 0 and 1. Depending on the document type, there might also be multimedia content. Such content can be described with categorisations and propositions in POLAR. Furthermore, documents and even annotations might be structured. For example,

```
d1[ 0.5 information  0.6 retrieval
    0.7 digital  0.3 libraries
    s1[ 0.4 information  0.2 retrieval ]
m1[ o1[] o2[]
    house(o1) tree(o2) o2.leftOf(o1) ]
```

states that $d1$ can be represented by the term propositions `information`, `retrieval`, `digital` and `libraries` with the corresponding probabilities (as the outcome of a text indexing process) and has a subsection $s1$ which is part of $d1$. $s1$ can be indexed with 'information' and 'retrieval'. Furthermore, a multimedia object $m1$ might be described by a categorisation of its components and spatial properties; in this example, it says that $m1$ has two components $o1$ and $o2$ which are a house and a tree, respectively, and $o2$ appears left of $o1$. With these mechanisms we are able to deal with structured, textual and multimedia documents and annotations. Note that with the information above we can automatically estimate termspace probabilities based on the inverse document frequency of terms, as it is discussed in [23, Section 4.2.1].

The content of an annotation is derived from its corresponding AnnotationBody instance. With Annotation($a1$), AnnotationBody($a1Body$) and hasAnnotationBody($a1, a1Body$) we assign the indexed content of $a1Body$ to $a1$, e.g.,

```
a1[ 0.6 search  0.8 big  0.7 issue ]
```

is a possible representation of a in POLAR if the content of $a1Body$ is something like "search is a big issue".

The representation of document content and structure in POLAR is the same as in POOL. To model annotation hypertexts and threads, we need to extend POOL, as we will see now.

3.1.3 Annotation Hypertexts and Threads in POLAR

We have discussed how the content of an annotation can be described in POLAR. Now we have to connect documents, fragments and annotations according to the given annotation hypertext.

3.1.3.1 Content Level Annotations.

We say `d[`p `*a]` if the content level annotation a annotates d, i.e. hasAnnotationTarget(a, d) and ContentLevelAnnotation(a). The *access probability* p is determined by the application. `d1[0.8 *a1]`, for example, means that document $d1$ is annotated by $a1$ and this annotation is accessed with 0.8 probability.

3.1.3.2 Meta Level Annotations.

Meta level annotations make assertions about objects on the meta level. A judgement $j1$ about a document $d1$ saying "this is a good introduction" would be modelled in POLAR as

```
d1[ 0.5 information  0.6 retrieval
    0.7 digital  0.3 libraries
    0.7 @j1 ]
j1[ 0.3 good  0.7 introduction ]
```

In general, `d[`p `@j]` means that there exists an annotation j which makes assertions about d on the meta level and is accessed from d with probability p. More formally, `d[`p `@j]` iff hasAnnotationTarget(j, d) and MetaLevelAnnotation(j).

3.1.3.3 Fragments.

When users create an annotation about a certain passage of a document, they first select the corresponding document fragment. This fragment is also a part of a document, and the fact that this was an annotation target should be expressed in our framework as well, since this knowledge can be valuable in the retrieval process. For example,

```
d1[ 0.5 information  0.6 retrieval
   0.7 digital  0.3 libraries
   0.8 f1|| 0.9 digital  0.5 libraries *a1|| ]
```

means that a fragment $f1$ of $d1$ which is about digital libraries was selected as an annotation target for $a1$; we refer to this fragment as an *annotated part* of $d1$. Formally, `d[`p `f||*a||])` iff isFragmentOf(f, d) and hasAnnotationTarget(a, f). Since annotated fragments and their corresponding annotations are closely related, they share the same access probability.

3.1.3.4 Merged Annotation Targets.

Experiments with email discussions have shown that annotation targets contain crucial information to determine what an annotation is about (see Section 4). In the following example

```
a1[ 0.8 t1< 0.5 information  0.6 retrieval
            0.7 digital  0.3 libraries >
    0.6 search  0.8 big  0.7 issue ]
```

$t1$ is the *merged annotation target* of $a1$ and is about information retrieval in digital libraries. More generally we say that an expression `a[`p `t<...>]` states that t is the merged annotation target of a and that this context is accessed with probability p. Merged annotation targets are constructed as follows. Let $target_a = \{o|\mathsf{hasAnnotationTarget}(a, o)\}$ denote the set of annotation targets of a. For a merged annotation target t of a, a proposition *prop* belongs to t iff there exists $o \in target_a$ such that *prop* belongs to o. This way we create a new document t which contains all propositions of a's annotation targets. In the indexing step, probabilities are calculated for each proposition, e.g. based on the term frequency and length of the newly created document t in case of terms. Since merged annotation targets are a kind of artificial documents created from the fact that they or their parts are annotated, these objects cannot be annotated any more but play a certain role in the retrieval process (as we will see later).

3.1.3.5 References.

References, which are also a component of annotation hypertexts, are syntactically represented in POOL as `o[<=a]`, which means object o is referenced by annotation a.

3.1.3.6 Polarity.

Another attribute of annotations we identified before is their polarity. The polarity of an annotation might be explicitly determined by, e.g., the annotation type. When an annotation type or a polarity is not explicitly given, machine learning algorithms could be applied to determine the polarity, similar to sentiment classification [21]. In this case we might gain uncertain knowledge about the polarity of an annotation which can be expressed assuming an open world as follows:

```
0.5 +-a1 [  0.6 search  0.8 big  0.7 issue ]
```

means a is positive with a probability 0.5

```
0.5/0.1 +-a1 [  0.6 search  0.8 big  0.7 issue ]
```

means a is positive with 0.5 probability and negative with a probability of 0.1.

3.1.3.7 Cycles.

We defined annotation hypertexts as being acyclic. This means we underly some restrictions when modelling an annotation hypertext in POLAR. To avoid cycles, direct or indirect mutual references like

```
a[*b]  b[*c] c[*a]
```

or

```
a[<=b] b[*c] c[<=a]
```

are forbidden.

3.2 Contexts and Knowledge Augmentation

3.2.1 Definition

The reason why we did not just model properties like $\mathsf{hasAnnotationTarget}(a, d)$ as a POLAR property `a.hasAnnotationTarget(d)` becomes clear when introducing the concept of *contexts* and *knowledge augmentation*. In POLAR, every object establishes a *context*. Within their context, objects have a specific knowledge determined by their content. For example, $a1$ and $d1$ establish a respective context (which we denote like the object establishing it). $d1$ knows about 'information', 'retrieval', 'digital' and 'libraries' in its context, but nothing more, whereas $a1$ knows about 'search', 'big' and 'issue'. Our representation `d[`p `*a]` also states that a is a *subcontext* of d which is accessed from d with probability p (the same holds true for annotated fragments and merged annotation targets). If we access a from d, we create an *augmented context* $d(a)$; in this augmented context, all propositions of d and a are known according to the proposition and access probabilities. For example, $d1(a1)$ knows about 'search' with a probability of $0.8 \cdot 0.6 = 0.48$.

In POLAR, knowledge augmentation is recursive; if in our example we have `a1 [... *a2]`, we first create an augmented context $a1(a2)$ and then finally $d(a1(a2))$. Formally, we create augmented contexts as follows. Let $prop_c$ be a proposition in a context c (precisely: the event that *prop* appears in c) . S_c is the set of subcontexts of c. For each context c we can uniquely identify an augmented context $c' = c(s'_1, \ldots, s'_n)$ with $n = |S_c|$ and s'_i as the augmented context of s_i. acc_s denotes the event that we actually access s from its supercontext. We calculate new probabilities $P(prop_{c'})$ for all propositions appearing in the augmented context c' as

$$P(prop_{c'}) = P\left(prop_c \vee \bigvee_{s \in S_c} (acc_s \wedge prop_{s'})\right)$$

The probability of combinations of probabilistic events is calculated as

$$P(e_1 \wedge \ldots \wedge e_n) = P(e_1) \cdot \ldots \cdot P(e_n)$$

and

$$P(e_1 \vee \ldots \vee e_n) = \sum_{i=1}^{n} (-1)^{i-1} \left(\sum_{\substack{1 \le j_1 < \\ \ldots < j_i \le n}} P(e_{j_1} \wedge \ldots \wedge e_{j_i}) \right)$$

(the latter one is the inclusion-exclusion formula).

3.2.2 Application in POLAR

The idea of knowledge augmentation has its roots in the scenario that if a user first reads a document d and then an annotation a, her knowledge is augmented accordingly. From an indexing and retrieval perspective, augmentation addresses two issues: first, the *vocabulary problem* – authors of documents and users seeking documents might use different terms to express the same concepts. In this scenario, we also deal with a third group, the annotators. The vocabulary problem might be decreased when the chances are higher that a document can be topically related to terms not contained in the document itself, but in annotations. Second, *topic emphasis*; we can raise the certainty that a document is about a topic if it has annotations which are about this topic as well. If, for example, users discussed a certain topic which also appears in the document, they associated the document with this topic, providing more evidence that the document is really about the given topic. Furthermore, knowledge augmentation might be a good tool to find *best entry points* in discussion threads, since they combine an

entity's knowledge with the one of its corresponding subthread. Knowledge augmentation is thus highly motivated for content level annotations.

For meta level annotation, knowledge augmentation is not defined. Meta level annotations do not influence the content of the object they belong to, but give a judgement about it. However, meta level annotations might be interesting for certain kinds of queries, as will be discussed in Subsection 3.3.

Similar to annotations, annotated parts also establish a context which should be considered for knowledge augmentation. Users annotating a fragment f of document d found this passage important enough to spend some time on it, even if the annotation is negative. Our claim is that such annotated passages are *implicitly highlighted* (this is especially true for fragments annotated with an annotation of class Highlighting, which are then explicitly highlighted). The more users annotate a specific passage (implicitly or explicitly), the more we get an *n-way-consensus* [19] that this passage has some value in it. Our hypothesis is that we thus receive additional evidence that the corresponding document should be indexed with the propositions (terms) in the annotated part, resulting in a higher probability of these propositions in $d(f)$. In our example, $f1$ is a subcontext of $d1$. $f1$ does not contribute any new propositions to $d1(f1)$, but the probabilities of the terms `digital` and `library` are raised in the augmented context $d1(f1)$ according to their probabilities in $d1$ and $f1$. Although an annotated fragment f constitutes a subcontext of its corresponding document d, it is permeable w.r.t. the annotation a belonging to f, meaning that a is still regarded as a direct subcontext of d, but not of f (in fact, fragments do not have any subcontext at all). Applied to our example, $d1$ has two subcontexts $f1$ and $a1$, and propositions of these subcontexts are directly propagated to $d1(f1, a1)$; but nothing is propagated from $a1$ to $f1$.

Merged annotation targets play a special role in the knowledge augmentation process. A merged annotation target t is only considered if the annotation a it belongs to establishes the highest supercontext of interest, but not if a plays the role of being a subcontext of some higher context d. This means if we want determine the augmented context $d(a)$, t and the augmented context $a(t)$ are not considered, but only a. If, on the other hand, we are interested in the augmented context of a and its subcontexts without the intention to further propagate the augmented values to a higher context (a is only supercontext, but not subcontext), $a(t)$ would be part of the augmentation. The reason for this strategy is that the merged annotation target t of a always contains knowledge from the supercontexts of a only (which is already reflected there), so we assume that an augmentation would not yield any new knowledge for these supercontexts[6]. The propositions `information`, `retrieval`, `digital` and `libraries` from $t1$ are propagated to $a1(t1)$ (this way we adhere to the fact that $a1$ is also about "information retrieval" (as a synonym for "search")), but they are *not* propagated to $d1(a1)$.

[6]Note that this is a simplified assumption. If a annotates two fragments $f1$ and $f2$, t contains terms from both fragments. Implicitly, $f1$ and $f2$ are somehow related due to the fact that they were target of the same annotation, and such a relation might possibly be subject to knowledge augmentation as well, but we neglect this for the time being.

3.3 Retrieval in POLAR

While the creation of the knowledge base described above is the outcome of a context-based indexing step, we are going to discuss possible retrieval options in POLAR now. *Queries* are expressed as headless rules, with the retrieval target in capital letters.

3.3.1 Database Queries

Database queries return facts from the given knowledge base. They are called "database queries" since they do not deal with any uncertain knowledge. Suppose that annotations have a property `author` denoting the author of an annotation. The query

```
?- A.author(turner) & annotation(A)
```

returns the URIs of turner's annotations. Similarly,

```
?- d1[ *A ]
```

returns all annotations annotating $d1$, whereas

```
?- D[ *a1 ] & document(D)
```

returns all documents annotated by $a1$. For references,

```
?- d1[ <=A ]
```

returns all annotations referencing $d1$ and

```
?- A[ <=a1 & <=a2 ] & comment(A)
```

returns all comments referenced by both $a1$ and $a2$.

3.3.2 Content-oriented Queries

Content-oriented queries deal with uncertain knowledge and calculate a retrieval status value (RSV) for each object w.r.t. the query and according to the probabilities of their propositions.

```
?- A[ big | issue ] & A.author(turner)
                    & annotation(A)
```

returns all of turner's annotations containing either 'big' or 'issue'; in our knowledge base, $a1$ would be assigned the RSV $0.8 + 0.7 - 0.8 \cdot 0.7$ (given turner is its author) since `big` and `issue` are seen as probabilistic events whose disjunction is calculated with the inclusion-exclusion formula.

```
?- D[ information & retrieval ] & document(D)
```

returns all documents about "information AND retrieval"; for $d1$, this leads to an RSV of $0.5 \cdot 0.6 = 0.3$ for the disjunction of events. No augmentation is applied when processing this query, but only the contexts established by the objects themselves would be considered. To tell the system to perform knowledge augmentation, we prefix our query with '`//`'.

```
?- //D[ information & search ]
```

yields $d1$ as well (while `?- D[ information & search ]` does not). Here we propagate the probability of 'search' to the augmented context $d1(a1)$ and retrieve d with a probability of $0.5 \cdot 0.48 = 0.24$

Besides knowledge augmentation, there is another retrieval strategy which we call *relevance augmentation*. In contrast to knowledge augmentation, we calculate the retrieval status value of each single object first (without performing any augmentation). Relevance augmentation then

means that we propagate the retrieval status value of a context to its supercontext to create the augmented context. As an example, when processing the query "information OR retrieval OR search",

```
?- <<D[ information | retrieval  | search ]
```

we first calculate retrieval status values for $d1$ and $a1$. For $d1$, we get a probability of $0.5 + 0.6 - 0.5 \cdot 0.6 = 0.8$ while for $a1$ we get 0.6 probability. To get the retrieval status value for the augmented context $d1(a1)$ we again consider the probability that $a1$ is accessed from $d1$ and compute $0.8 + 0.8 \cdot 0.6 - 0.8 \cdot 0.8 \cdot 0.6 = 0.896$. The prefix '<<' tells the system to perform relevance augmentation. Relevance augmentation is a suitable strategy for queries involving meta level annotations. For example,

```
?- <<D[ digital & libraries
        @[ good & introduction ] ]
```

retrieves all good introductions about digital libraries (but not objects which contain the terms 'good' and 'introduction'). To compute the retrieval status value of $d1$ for this query with relevance augmentation, we calculate the RSV of $j1$ first and propagate this value to the augmented context $d1(j1)$.

The query

```
?- D[ || digital & libraries || ] & document(D)
```

returns documents whose annotated parts are about digital libraries. This can be useful if users only remember that they annotated a passage with the given content in the desired document. Finally, the query

```
?- F|| digital & libraries ||
```

returns only fragments being annotated parts about digital libraries. This kind of query enables direct access to annotated document fragments in case users are only interested in these parts.

The queries presented so far all dealt with positive annotations. The question is how we can handle negative annotations w.r.t. knowledge and relevance augmentation. [11] describes mechanisms for dealing with negative evidence and relevance augmentation based on four-valued probabilistic Datalog. Here, positive annotations have an effect on the probability that a document is relevant, whereas negative annotations raise the probability that a document is not relevant. With respect to knowledge augmentation, the proposition a might be propagated as $\neg$a in the augmented context for negative annotations. The incorporation of negative evidence is subject to further discussion.

3.4 Implementation Notes

We introduced POLAR as a framework for annotation-based information retrieval using an object-oriented, logic-based representation to allow for flexible and complex queries. While POLAR itself has not been implemented yet, the ideas of knowledge and relevance augmentation were partly realised using four-valued probabilistic datalog (FVPD). For example, [11] describes a relevance augmentation approach considering content level annotations with polarity, applied within the COLLATE annotation model. On the other hand, [10] realises knowledge augmentation for annotated parts (called "highlight quotations" there) and merged annotation targets (called "context quotations") to evaluate new concepts of discussion search in emails threads. Our strategy to implement POLAR is to translate its expressions to semantically equivalent ones in FVPD, as it was done with POOL where knowledge augmentation is applied for structured documents [23]. We will then combine the augmentation strategies presented in [11, 10] with the additional ones discussed in this section. With HySpirit[7] [13] and pDatalog++ [20], two implementations of (four-valued) probabilistic datalog exist which are possible backbones for POLAR. Main implementation efforts would focus on translating POLAR programs into FVPD and execute them with one of the appropriate engines.

4. APPLICATION EXAMPLE AND EXPERIMENTAL RESULTS

In this section we discuss an example of a POLAR representation using email discussions. We will also present some experimental results which motivate our model described before. Our example collection consists of emails as they can be found in the W3C email lists[8]. The reason why we chose this collection is that it was used in the latest TREC Enterprise Track (TRECent), where we evaluated some of our concepts, in particular knowledge augmentation with annotated parts and merged annotation targets. Furthermore, email discussions and annotations share the same characteristics and are therefore a good testbed to validate our concepts[9]. In fact, a digital library might support textual annotation mechanisms which are similar to emails.

4.1 Emails as Annotations

In the left part of Fig. 3 we can see that emails consist of several parts, in particular

- the *quotations*, which are passages of the original text. Quotations are identified by quotation characters like '>' or ':' which prefix each line of the quotation; combinations of them usually define the *quotation depth*. In m3, quotations belonging to m1 have the depth 2 and are identified by two quotation characters ('> >'), whereas quotations belonging to m2 are identified by the single quotation character '>';
- the *new* part containing new content by the author of the message.

In replies we find many of the characteristics of annotations introduced in Section 2. The new part contains annotations (here: textual, shared comments) of passages of previous messages (and can thus be seen as the annotation body); quotations and the quotation depth determine the annotation targets. The right part of Fig. 3 illustrates this. Based on these considerations, we will now discuss how to represent email threads in POLAR.

4.2 Representation of Email Threads in POLAR

We take the discussion thread in Fig. 3 as an example. The root of an email discussion thread does not reply to

[7] http://qmir.dcs.qmul.ac.uk/hyspirit.php
[8] http://lists.w3c.org/
[9] Due to the lack of a "real" digital library annotation test collection, we had to rely on the TRECent collection.

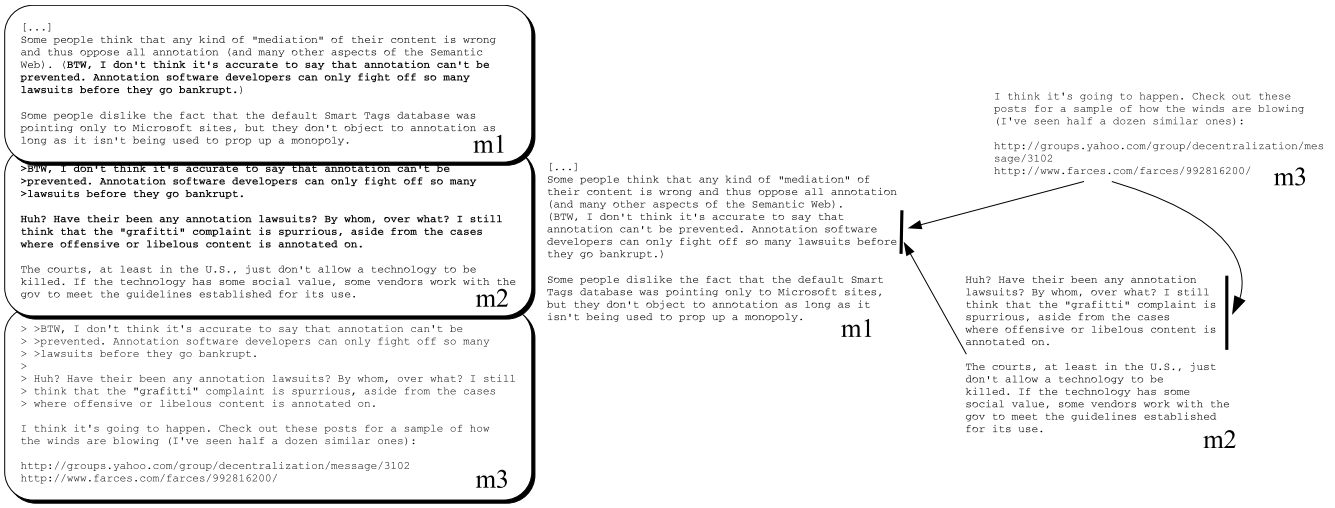

Figure 3: An email discussion thread and its view as annotations

any other email (in theory) and can therefore not be regarded as an annotation in our model. To reflect this, we introduce new classes Message $\sqsubseteq$ AnnotatableObject and Reply $\sqsubseteq$ Annotation $\sqcap$ Message and categorise our messages accordingly:

```
annotation(A) :- reply(A)
message(M) :- reply(M)
annotatableObject(E) :- message(E)
message(m1) reply(m2) reply(m3)
```

Message m1 can be represented in POLAR as follows (we only consider a part of the term set of messages; all probabilities are fictitious):

```
m1[ 0.5 mediation 0.6 annotation 0.5 lawsuit
    0.6 people 0.4 database
    0.3 a_m1_m2|| 0.7 annotation 0.6 lawsuit *m2 ||
    0.3 a_m1_m3|| 0.7 annotation 0.6 lawsuit *m3 ||
  ]
```

This says that the body of m1 consists of terms like 'mediation', 'annotation', 'lawsuit', 'people' and 'databases'. A fragment of m1 containing the terms 'annotation' and 'lawsuit' is quoted by two other emails, m2 and m3 (and is therefore their annotation target). The probability that we access the context a_m1_m2 or a_m1_m3 to determine the augmented context m1(a_m1_m2) or m1(a_m1_m3), respectively, is 0.3. m2 and m3 can be represented as

```
m2[ 0.9 q_m2< 0.75 annotation 0.7 lawsuit >
    0.55 annotation 0.5 lawsuit 0.3 graffiti
    0.4 courts 0.6 technology
    0.3 a_m2_m3|| 0.8 annotation 0.75 lawsuit
                  0.6 graffiti *m3 || ]

m3 [ 0.9 q_m3< 0.85 annotation 0.8 lawsuit
               0.5 graffiti >
     0.4 happen 0.6 posts 0.3 sample 0.55 winds ]
```

The quotation of m2, q_m2, contains the terms 'annotation' and 'lawsuits', whereas the quotation of m3, q_m3, additionally contains the term 'graffiti'. q_m3 is a merged annotation target from a_m1_m3 and a_m2_m3. In this model we applied a simplification of emails. Emails usually consist of several quotations and new parts. We merge all quotations and all new parts, so as a result we gain one big quotation and one big new part for each email. A fine-grained model would regard each new part as an annotation of the corresponding quotation. Since in the experiments which we are going to discuss now whole emails where the retrieval targets, we apply this simplification for the time being.

4.3 Experimental Results

Our group participated in the 2005 TREC Enterprise Track[10] in the discussion search task where relevant emails had to be found [10]. The test collection consisted of the W3C email lists described above, with 174,307 emails and 59 test queries. In our experiments we did not represent

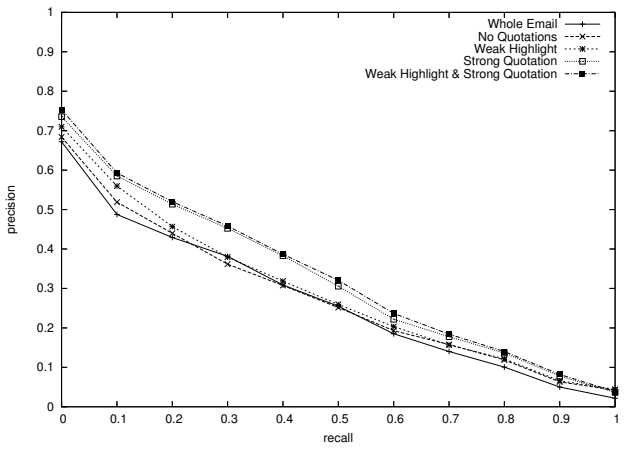

Figure 4: Recall-precision graph of selected results

emails in POLAR, but used an equivalent representation in probabilistic datalog so we could use HySpirit for our test

[10] http://www.ins.cwi.nl/projects/trec-ent/

runs. Document-term probabilities were estimated using a term-frequency based measure, whereas query terms were weighted using their inverse document frequency. Please refer to [10] for a detailed description of our setup. The "Whole Email" experiment regarded a whole email (quotations and new part) as a single message, without any augmentation. In the "No Quotations" run (and all other experiments), a message consisted only of the new part of an email. In this experiment, no augmentation was performed, so the RSV of a message was computed based on its content only. These two experiments served as baselines for us. As discussed before we regard the fact that a passage of a document was annotated as a kind of implicit highlighting of this passage, so annotated parts are regarded as highlighted parts. In the "Weak Highlight" experiment we evaluated our hypothesis that augmenting the knowledge of an annotation context with those of its highlighted (annotated) parts enhances retrieval effectiveness. All annotated fragments were assigned a global access probability of 0.3, so they were only weakly connected to their corresponding object. The goal of the "Strong Quotation" run was to evaluate knowledge augmentation with merged annotation targets (the quotations in case of an email). The global access probability was 0.9, so quotations were strongly connected to new parts. Finally, in "Weak Highlight & Strong Quotation" we performed knowledge augmentation with both annotated (highlighted) fragments and merged annotation targets with global access probabilities of 0.3 and 0.9, respectively.

Figure 4 shows the interpolated recall-precision graph of some selected results and Table 1 shows further results, mean average precision (MAP) and the precision when the first 10 documents are considered (P@10). We regard these results as a justification of our augmentation strategies discussed in Section 3, although we did not evaluate the propagation of the content of an annotation to the annotated object yet. We also performed several other runs with different combinations of access probabilities of 0.3 and 0.9, respectively, which did not yield better results. Future evaluations have to show how other access probabilities perform w.r.t. retrieval effectiveness. Refer to [10] for a more thorough discussion and the beforementioned additional results.

Run	MAP	P@10
Whole Email	0.2565	0.3966
No Quotations	0.2599	0.4102
Weak Highlight (WH)	0.2726	0.4458
Strong Quotation (SQ)	0.3094	0.4627
WH & SQ	0.3174	0.4881

Table 1: Further results

5. RELATED WORK

Our reflections on annotations and their characteristics is highly influenced by the existing annotation systems we mentioned in Section 2 [4, 7, 16, 17, 24]. They are examples for the distinct concepts of annotations they address, but certainly not the only systems dealing with annotations. The Multivalent Browser [22] is another interesting tool which allows for creating in-situ annotations in different document formats (like Postscript, PDF, ASCII). The nature of annotations, their role and application in digital libraries and collaboratories and possible annotation-based retrieval functions are studied and discussed in, e.g., [3, 1, 19, 18].

Golovchinsky *et al.* use annotations for relevance feedback by only considering highlighted terms instead of whole relevant documents for document search [14]. Experiments show a gain in retrieval effectiveness against classic relevance feeback. Agosti and Ferro see annotations as context for document search and describe retrieval methods based on HIR and data fusioning [2]. Their algorithm computes two result sets, based on documents alone and on their corresponding annotations, respectively. These result sets and the RSVs are fusioned to get the final result.

Annotation-based retrieval has a strong relation to Hypertext Information Retrieval (HIR) [5] due to the fact that we deal with annotation hypertexts. An example numeric HIR algorithm based on spreading activation, which is able to cope with typed and negative links, is found in [9].

An interesting feature-based approach for discussion search is reported in [27] by Xi *et al.* They define several content, thread-tree and author features. Their results show that using the thread context for discussion search improves retrieval effectiveness. Xi *et al.* neglect quotations from previous messages. Several other discussion search approaches were presented at the TREC 2005 conference [26].

6. CONCLUSION

In this paper we introduced POLAR, a probabilistic object-oriented logical framework for representing annotation hypertexts. Based on our observations of existing annotation models, we created an object-oriented view on digital objects, documents and annotations, and their relationships. This view was the foundation for defining the concepts of POLAR, which is able to represent annotation hypertexts and specific characteristics of annotations, especially (merged) annotation targets and annotated fragments. We also motivated and discussed knowledge and relevance augmentation which are core concepts in annotation-based retrieval strategies. First experiments with email discussions indicate that the definition of POLAR is reasonable for the task of annotation-based information retrieval.

Future work will focus on a full implementation of POLAR based on existing implementations of four-valued probabilistic datalog and on additional evaluations of knowledge and relevance augmentation.

7. ACKNOWLEDGEMENTS

Our work is partially funded by the DELOS Network of Excellence on Digital Libraries (URL: http://www.delos.info/), as part of the Information Society Technologies (ISO) Programme of the European Commission (Contract G038-507618).

8. REFERENCES

[1] M. Agosti and N. Ferro. Annotations: Enriching a digital library. In Constantopoulos and Sølvberg [8], pages 88–100.

[2] M. Agosti and N. Ferro. Annotations as context for searching documents. In F. Crestani and I. Ruthven, editors, *Information Context: Nature, Impact, and Role: 5th International Conference on Conceptions of*

Library and Information Sciences, CoLIS 2005, volume 3507 of *Lecture Notes in Computer Science*, pages 155–170, Heidelberg et al., June 2005. Springer.

[3] M. Agosti, N. Ferro, I. Frommholz, and U. Thiel. Annotations in digital libraries and collaboratories – facets, models and usage. In Heery and Lyon [15], pages 244–255.

[4] M. Agosti, N. Ferro, and N. Orio. Annotating illuminated manuscripts: an effective tool for research and education. In *JCDL '05: Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 121–130, New York, NY, USA, 2005. ACM Press.

[5] M. Agosti and A. F. Smeaton, editors. *Information Retrieval and Hypertext*. Kluwer Academic Publishers, Boston et al., 1996.

[6] F. Baader, D. Calvanese, D. McGuiness, D. Nardi, and P. Patel-Schneider, editors. *The Description Logic Handbook – Theory, Implementation and Applications*. Cambridge University Press, Cambridge, UK, 2003.

[7] P. Bottoni, R. Civica, S. Levialdi, L. Orso, E. Panizzi, and R. Trinchese. Madcow: a multimedia digital annotation system. In *AVI '04: Proceedings of the working conference on Advanced visual interfaces*, pages 55–62, New York, NY, USA, 2004. ACM Press.

[8] P. Constantopoulos and I. T. Sølvberg, editors. *Research and Advanced Technology for Digital Libraries. Proc. European Conference on Digital Libraries (ECDL 2003)*, Lecture Notes in Computer Science, Heidelberg et al., 2003. Springer.

[9] H. P. Frei and D. Stieger. The use of semantic links in hypertext information retrieval. *Information Processing and Management*, 31(1):1–13, Jan. 1994.

[10] I. Frommholz. Applying the annotation view on messages for discussion search. In Voorhees and Buckland [26].

[11] I. Frommholz, U. Thiel, and T. Kamps. Annotation-based document retrieval with four-valued probabilistic datalog. In T. Roelleke and A. P. de Vries, editors, *Proceedings of the first SIGIR Workshop on the Integration of Information Retrieval and Databases (WIRD'04)*, pages 31–38, Sheffield, UK, 2004.

[12] N. Fuhr, N. Gövert, and T. Rölleke. DOLORES: A system for logic-based retrieval of multimedia objects. In W. B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson, and J. Zobel, editors, *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 257–265, New York, 1998. ACM.

[13] N. Fuhr and T. Rölleke. HySpirit – a probabilistic inference engine for hypermedia retrieval in large databases. In *Proceedings of the 6th International Conference on Extending Database Technology (EDBT)*, pages 24–38, Heidelberg et al., 1998. Springer.

[14] G. Golovchinsky, M. N. Price, and B. N. Schilit. From reading to retrieval: freeform ink annotations as queries. In *Proceedings of the 22nd International Conference on Research and Development in Information Retrieval*, pages 19–25, New York, 1999. ACM.

[15] R. Heery and L. Lyon, editors. *Research and Advanced Technology for Digital Libraries. Proc. European Conference on Digital Libraries (ECDL 2004)*, Lecture Notes in Computer Science, Heidelberg et al., 2004. Springer.

[16] J. Kahan, M. Koivunen, E. Prud'Hommeaux, and R. Swick. Annotea: An open RDF infrastructure for shared web annotations. In *Proceedings of the WWW10 International Conference*, Hong Kong, May 2001.

[17] C.-P. Klas, N. Fuhr, and A. Schaefer. Evaluating strategic support for information access in the DAFFODIL system. In Heery and Lyon [15].

[18] C. Marshall and A. Brush. Exploring the relationship between personal and public annotations. In *JCDL '04: Proceedings of the 4th ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 349–357, New York, NY, USA, 2004. ACM Press.

[19] C. C. Marshall. Toward an ecology of hypertext annotation. In *Proceedings of the ninth ACM conference on Hypertext and hypermedia: links, objects, time and space – structure in hypermedia systems*, pages 40–49, 1998.

[20] H. Nottelmann. PIRE: An extensible IR engine based on probabilistic datalog. In D. E. Losada and J. M. F. Luna, editors, *27th European Conference on Information Retrieval Research (ECIR 2005)*, 2005.

[21] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In *Proc. of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 79–86, 2002.

[22] T. A. Phelps and R. Wilensky. Multivalent annotations. In C. Peters and C. Thanos, editors, *Research and Advanced Technology for Digital Libraries. Proc. European Conference on Digital Libraries (ECDL 1997)*, volume 1324 of *Lecture Notes in Computer Science*, pages 287–303, Heidelberg et al., 1997. Springer.

[23] T. Rölleke. *POOL: Probabilistic Object-Oriented Logical Representation and Retrieval of Complex Objects*. PhD thesis, University of Dortmund, Germany, 1998.

[24] U. Thiel, H. Brocks, I. Frommholz, A. Dirsch-Weigand, J. Keiper, A. Stein, and E. Neuhold. COLLATE - a collaboratory supporting research on historic european films. *International Journal on Digital Libraries (IJDL)*, 4(1):8–12, 2004.

[25] C. J. van Rijsbergen. A non-classical logic for information retrieval. *The Computer Journal*, 29(6):481–485, 1986.

[26] E. M. Voorhees and L. P. Buckland, editors. *The Fourteenth Text REtrieval Conference (TREC 2005)*, Gaithersburg, MD, USA, 2005. NIST.

[27] W. Xi, J. Lind, and E. Brill. Learning effective ranking functions for newsgroup search. In K. Järvelin, J. Allen, P. Bruza, and M. Sanderson, editors, *Proceedings of the 27st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 394–401, New York, 2004. ACM.