POLIS:
A Probabilistic Logic for Document Summarisation

.
.
.

Keywords: Information retrieval, logic-based retrieval, summarisation logic, structured document retrieval.

Abstract: Summarisation is an important and re-occurring task to be solved in manifold search applications and customer-specific scenarios. Therefore, we propose and investigate in this paper a new approach to summarisation, namely an approach to describing summarisation approaches in a new abstraction: a probabilistic logic for information summarisation (POLIS). POLIS features the usual advantages of abstraction such as robustness, flexibility, and, most importantly, re-usability. The research achievement relevant to information retrieval is on one hand the well-defined probabilistic semantics applying possible worlds semantics, and, on the other hand an implementation of POLIS where we take advantage of an existing probabilistic algebraic approach to IR, and prove applicability and investigate retrieval quality in large-scale experimental settings.

1 INTRODUCTION

Research in information retrieval is motivated by the need for methods which allow searchers to find documents relevant for their information needs in the large databases of documents in organisations and companies, and even larger collections of documents available online. The importance of efficient automatic document summarisation approaches as a tool for handling data was recognised as early as 1960 (Luhn, 1958),(Edmundson, 1969).

Document summaries can be either *extracts*, consisting of material verbatim present in the original document, or *abstracts*, coherent documents generated from –but not present verbatim in– the source document. The generation of abstracts is usually a postprocessing step following the extraction of “extract-worthy” material. Over the years, several methods where proposed for extracting the most salient information from documents: Sentence scoring strategies try to calculate a “score” for each document sentence, based on term- and sentence-frequencies in documents; heuristic sentence scoring strategies, incorporating information about the relative position of sentences within paragraphs; scores for “bonus” and “malus” words and information about sentence length. Natural Language Processing approaches try to determine the informativeness of bits of documents based on the syntactical structure of documents, and individual user background. Machine learning approaches attempt to determine the most relevant parts of documents by learning from given examples – documents with their associated, human generated abstracts or extracts.

Furthermore, probabilistic summarisation models based on the above strategies were developed, and some of those models were implemented in standalone summarisers, which can be included in other information retrieval systems (e.g. SUMMARIST (Hovy and Lin, 1997)). However, both (i.e. mathematical models and complete summarisation systems) represent extremes of approaches to document summarisation: probabilistic models define how a summary should be constructed, but need to be implemented. Summarisers, on the other hand, represent a “black box” approach to summarisation, where user interaction and control is minimal.

The work presented here aims to tread a middle ground: a summarisation *logic*, that allows experienced users to express probabilistically *how* a summary should be generated, without having to implement the summarisation models.

For most of the research on document summarisation carried out over the past decades, data to be summarised was largely flat and unstructured. Languages that were able to provide additional markup to signify structural information for documents were either considered too complex to be generally applicable (e.g. SGML), or were used for layout markup, rather than structural markup (e.g. HTML). Structural information about the documents to be summarised had to be extracted by the summarisation approaches manually.

The adoption of XML as a W3C standard for information repositories and data exchange allowed the inclusion of structural information into documents in an easy and flexible way. While the data-centric view of XML documents uses markups as a data interchange format, the document-centric view uses markup mainly for representing a document's logical structure (see e.g. (Fuhr et al., 2005)). The summarisation logic presented here makes use of the structural information present in XML and other structured documents, to allow users to specify which parts of documents should be extracted to form the summary.

The remainder of this paper is structured as follows: Section 2 provides an overview of past summarisation approaches. Section 3 shows how to model surface-level summarisation approaches in POLIS. POLIS is a probabilistic summarisation logic inspired by description logics, for which we define a syntax that borrows from XPath notation (subsection 3.5) (Clark and DeRose, 1999), while we base the semantics on Kripke structures (subsection 3.6). Section 4 details how we evaluated the summarisation logic, based on the AQUAINT corpus and the evaluation measures provided by the Document Understanding Conference (DUC). The results of our experiments are given in subsection 4.2. Section 5 concludes the discussion.

2 RELATED WORK

Document summarisation approaches can be broadly classified into two categories: approaches that incorporate natural language processing techniques (NLP), and statistical approaches rooted in information retrieval. While the former try to select most salient sentences in a document by assigning meaning to individual parts of documents and sentences, the latter take a frequentist point of view, where the importance of sentences is determined by term distributions and heuristics about the location of sentences in documents. The present discussion will focus on IR approaches only, and will mention NLP approaches only where there is an overlap with the IR approaches.

In early work on summarisation, the importance of sentences was determined by heuristic features, such as term frequencies in sentences, the location of sentences in the document, and sentence length. An overall score for sentences was derived by a linear combination of individual features (Luhn, 1958),(Edmundson, 1969). This early work exposed that individual features were not successful in determining the most salient sentences, however, different linear combinations of features correctly identified important concepts in texts. In later work, the combination of features was optimised using machine learning systems, that would derive ideal weighted combinations of features from a given knowledge base, consisting of full texts, and human generated extracts (Kupiec et al., 1995).

In later research, the focus shifted from term weighting strategies based on linear combinations of features to probabilistic sentence scoring strategies. For example, Saravanan *et al.* (Saravanan et al., 2005) present a multi-document summariser based on a K-mixture model for term distribution. Knight and Marcu (Knight and Marcu, 2002) present a probabilistic model for sentence compression, which goes beyond conventional probabilistic models for sentence extraction trained on given document / summary pairs.

While probabilistic summarisation models provide a high level of detail to users, they also require anyone wishing to summarise documents according to one of the models to implement that model, and to control every aspect of a summarisation system, from document parsing and representation, to the actual summary generation.

The other extreme are complete summarisation systems, such as SUMMARIST (Hovy and Lin, 1997), which implement complex summarisation models that take into account several sentence scoring strategies to detect the most salient parts of documents. The output of such summarisation systems is either an extract of the input document(s), or some conceptual representation for the later generation of summaries (e.g. SUSY (Fum et al., 1985), TOPIC (Reimer and Hahn, 1988), SCISORS (Rau et al., 1989)) Although such systems try to combine the best summarisation strategies available at any one time, they represent a black-box approach to summarisation, as the inner workings might not be accessible to a user, and the process of summary generation is fixed and cannot be tailored towards user needs.

Summarisation of structured documents adds another level of complexity to the summarisation task, as the explicit structure provides additional information. Recent approaches to structured document summarisation (here: XML summarisation) have mainly taken a data-centric point of view, where summarisation is seen as a form of data compression. Alam *et al.* (Alam et al., 2003) present a summarisation approach in which the original structure of the original document is retained. Litkowski tested a summarisation system for structured documents at the Document Understanding Conference (DUC) (e.g. (Litkowski, 2004)). However, neither of these systems makes it explicitly clear *how* sentences are scored, and how structure is used in the summarisation process.

The summarisation logic presented here provides a level of abstraction between explicit summarisation models and complete summarisation systems. It takes a document-centric point of view for the summarisation of structured documents, and has a well-defined semantics based on a possible worlds interpretation of structured documents (Fagin et al., 1995).

3 POLIS

3.1 Introduction

Summarisation is an important and re-occurring task in information retrieval and information retrieval systems. However, summarisation is usually an integral (non-reusable) part of a retrieval system, though the task re-occurs in many search systems. Therefore, we propose to model the task in a high abstraction layer (probabilistic logic), so that we can apply the summarisation of documents to many IR systems / environments.

In our model, summarisation becomes another abstraction layer, which is decoupled from the physical implementation of an information retrieval system. Furthermore, the specification of how summaries should be generated is isolated from the actual implementation of the summariser. This allows for our logic to be applied to both new and existing information retrieval systems, as well as an easy customisation of the generated summaries to IR systems users' needs.

The remainder of this section will detail how our summarisation logic is similar to existing IR description logics, followed by possible application scenarios for a summarisation logic. The section concludes with an exposition of a syntax and an introduction to the semantics.

3.2 Comparison to Description Logics

The purpose of a summarisation logic is to provide an abstraction layer for the task of document summarisation, such that summarisation models (and summaries) are *described* rather than implemented. For the task of document retrieval similar abstraction layers have been defined using description logics specifically designed for this purpose (Meghini et al., 1993). Since our summarisation logic shares some concepts with those description logics, we will introduce concepts common to both logics in this subsection, and will also highlight differences between the logics. This comparison is also presented in Table 1.

3.2.1 Syntax

Description logics specify the properties of individuals (or constants) in a domain of discourse, which in the context of information retrieval usually denotes documents. These constants participate in either monadic *concepts*, or dyadic *roles*. Roles provide information about the relationship between individuals. For example, the role "Author" details that a person "A" is the author of a paper "P". Monadic concepts give information about single individuals, and include ordinary truth valuations (i.e. *true* and *false*, sometimes denoted *top* and *bottom*), negation, and the assignment of identifiers to constants. More interesting for the purpose of a summarisation logic are additional *quantifiers*. These include – in addition to existential and universal quantification – *numbered* quantification. These are denoted by keywords such as *at_most*, *at_least*, or *exactly*, followed by a concept. Using these definitions, it is possible to define new concepts in terms of other, known concepts. For example, it is possible to state that a dog is an animal with exactly four legs using a statement like:

$$dog \doteq (\text{ animal } \textbf{and} \; (\textbf{exactly} \; 4 \text{ leg })).$$

From this one can see how it would be possible to formulate a (very simple) definition for a summary:

$$summary \doteq (\textbf{ exactly } 4 \text{ sentence }).$$

However, our summarisation logic explicitly uses the structure of documents, which would be difficult to achieve using the above kind of syntax. Instead, we use a syntax based on XPath –used more commonly for accessing structured documents–, with an added support for numbered quantification. This syntax will be shown informally in subsection 3.3, and will be introduced formally in subsection 3.5.

3.2.2 Semantics

Whereas the syntax of POLIS is different to common description logics, its semantics is similar to that commonly employed by DLs. It is common for DLs to use *denotational semantics*, or *possible worlds semantics*.

Table 1: Comparison of common DLs and POLIS.

	DLs	POLIS
Syntax	no syntax common to all DLs, usually inspired by predicate logics, no explicit support for structured documents	XPath based syntax + numbered quantification
Semantics	possible worlds semantics	possible worlds semantics

Such possible worlds semantics are usually expressed using Kripke structures. A Kripke structure is a tuple $M = (\mathcal{S}, \pi, \mathcal{R})$, where $\mathcal{S}$ is a set of states, π is a function on $\mathcal{S}$ that yields a truth value assignment for a proposition Φ in state s, such that $\pi(s) : \Phi \rightarrow \{\mathit{true, false}\}$, and $\mathcal{R}$ is a set $\{\mathcal{R}_1 \cdots \mathcal{R}_a\}$ of binary relations on $\mathcal{S}$. These relations are the so called possibility relations. For a tuple $(s_i, s_{i+1}) \in \mathcal{R}_i$, an agent a_i in state s_i considers s_{i+1} a possible state. For the present purpose, we will restrict the discussion to a two-valued logic, where propositions (e.g. terms) are either true or false. To accommodate the open-world assumption usually adopted in IR, section 3.6.2 will sketch an extension to a three-valued logic.

The above non-probabilistic interpretation structure can be extended by a function $\mathcal{P}$, that assigns to each agent a and each state s a probability space $(\mathcal{S}_{a,s}, \mathcal{H}_{a,s}, \mu_{a,s})$, where $\mathcal{S}_{a,s}$ is a subset of $\mathcal{S}$, $\mathcal{H}$ is a σ-algebra of $\mathcal{S}_{a,s}$, and μ is a probability mass function. This results in a probabilistic interpretation structure $M = (\mathcal{S}, \pi, \mathcal{R}, \mathcal{P})$ (Fagin and Halpern, 1994). With this interpretation structure, it is possible to define an interpretation function "$\models$". A common notation is $(M, s) \models \varphi$ if a proposition or formula φ is true at state s in the interpretation structure M. For a two-valued logic, the interpretation of propositions is defined as

$$\begin{array}{lcl} (M,s) \models_t \varphi & \iff & \pi(s)(\varphi) = \text{true} \\ (M,s) \models_f \varphi & \iff & \pi(s)(\varphi) = \text{false} \end{array}$$

This concludes the discussion of details of the semantics common to both POLIS and DLs. For the later discussion of POLIS semantics extensions , we additionally introduce two pieces of "notational sugar": we will use $\models$ as shorthand notation for $\models_t$ unless stated otherwise. Furthermore, we will frequently refer to the set of states an agent considers possible given the state it is in. For this we define a function $\mathcal{R}$, which returns for agent i in state s all those states considered possible: $\mathcal{R}_i(s) := \{s' \mid (s, s') \in \mathcal{R}_i\}$.

3.3 Sample Applications of POLIS

In this subsection, we show how a summarisation logic can be applied to existing information retrieval tasks. As a first example, consider a web information retrieval system. Here, the set of retrieved documents (web pages) needs to be displayed to a user in such a way as to be indicative of the content, to give users the ability to assess a document's relevance to the specified information need. User studies conducted by Wolf et al. show that more sophisticated summaries of documents retrieved in response to a user query reduce the time it takes for participants to complete given tasks (Wolf et al., 2004). Wolf et al. also note that their "work demonstrates that using information about the subcomponent structure of documents to guide selective extraction can result in more useful document summaries" (Wolf et al., 2004). This feature is directly supported by POLIS, as the explicit specification of the granularity at which document components should be extracted for summarisation is given by a POLIS expression. It is thus possible to not only summarise documents in response to a user query, but to also incorporate user preference in the summarisation model. For example, consider a retrieval system, where each returned document is represented by up to three most important sentences occurring in it. An initial idea on how to allow a user to express this could simply look like

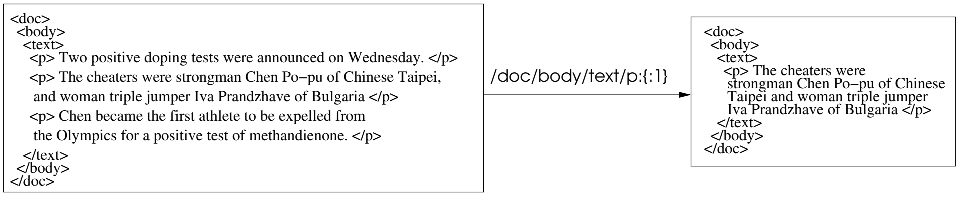
/sentence:<=3.

Processing such an instruction requires a fair amount of insight into the structure of documents. This could be achieved through the use of schema mappings from user input to the actual structure of documents, however, this level of abstraction is currently not achievable in POLIS. The actual POLIS expression to carry out this operation is:

/*/body/sec/p/s:{at_most 3},

for a document that has sentences as parts of paragraphs, which are children of section elements inside 'body' elements.

As a second example, consider the task of expert profiling. It is possible to view the profiling of experts as a form of summarisation. Here, the task is to derive from a knowledge base, e.g. a collection of emails, a good representation of the knowledge of an expert. Summarising an expert by extracting the ten most important documents associated with that expert could be expressed as **/expert/docs:10**. Here, the actual PO-



```
<doc>
 <body>
  <text>
   <p> Two positive doping tests were announced on Wednesday. </p>
   <p> The cheaters were strongman Chen Po−pu of Chinese Taipei,
    and woman triple jumper Iva Prandzhave of Bulgaria </p>
   <p> Chen became the first athlete to be expelled from
    the Olympics for a positive test of methandienone. </p>
  </text>
 </body>
</doc>
```

```
<doc>
 <body>
  <text>
   <p> The cheaters were
    strongman Chen Po−pu of Chinese
    Taipei and woman triple jumper
    Iva Prandzhave of Bulgaria </p>
  </text>
 </body>
</doc>
```

Figure 1: A sample document, its one-sentence summary, and the POLIS expression to generate it.

LIS expression for summarising an email might look as follows:

/doc/body/*:$\{$**at_most 4**$\}$,

meaning “retrieve the four most important elements, which are children of a ’body’ element, which is a child of a ’Doc’ element.”.

Both of these applications use a summarisation logic in a post-processing stage. However, such a logic might also be applied in a preprocessing stage, for example to limit the content to be used for indexing in both granularity and size. The use of summaries as substitutes for indexing has been explored by Sakai and Sparck-Jones (Sakai and Sparck-Jones, 2001), where it was shown that summary indexes perform as good as full-content indexes, provided the summaries are of sufficiently high quality.

The process of generating a document summary defined by a POLIS expression is shown in figure 1.

3.4 Probabilistic Model

The probabilistic model implemented by POLIS is motivated by the sentence scoring strategies of earlier summarisation approaches, as outlined in the introduction. The task of any summarisation system is to extract the most salient textual elements from either a single document, or a collection of documents; the extracted elements should be as representative as possible of their origin. We will refer to this property as the “aboutness” of textual elements; any textual element that participates in the formation of a summary should be as much “about” its source document(s) as possible. It follows that the probability of any (random) text element being included in a summary should be proportional to the degree that that textual element is “about” its surrounding document or context.

With a view on general research in IR, the aboutness of textual elements to their surrounding contexts is very similar to the idea that documents are “about” queries. In traditional IR, the relevance estimation between a document d and a query q can be seen as “the extent to which q might be inferred from d” (van Rijsbergen, 1986), expressed as $d \rightarrow q$, where $\rightarrow$ is not the material implication $\supset$ (defined as $d \supset q = \neg q \vee d$), but rather has a probabilistic interpretation such that the correspondence (or aboutness) of d and q is expressed as the degree to which $d \rightarrow q$ is true. Thus, aboutness(d,q) := P$(d \rightarrow q)$. Following van Rijsbergen, the probability P$(d \rightarrow q)$ can be defined as the conditional probability P$(q|d)$, which can be rewritten as $\frac{P(d \cap q)}{P(d)}$. Assuming term independence, the joint probability P$(d \cap q)$ and the document probability P(d) can be calculated as the total probability over a set of disjoint terms, with P$(d \cap q) = \sum_t P(d \cap q|t)P(t)$ and P$(d) = \sum_t P(d|t)P(t)$. Using these definitions, it is possible to rewrite P$(d \rightarrow q)$ as $\sum_t P(t|d)P(q|t)$. In terms of traditional IR models, P$(t|d)$ can be interpreted as the term-frequency *tf*, while P$(q|t)$ is a measure of term specificity, given by the collection frequency weight (CFW), or inverse document frequency *idf*. The sentence weighting model implemented by POLIS directly builds upon this IR model. Terms in textual elements occurring at the granularity specified by a user are weighted using the *idf* collection weight, while terms in the surrounding document or context are weighted using the document frequency *tf*. The sentence weighting model thus implements a measure of “aboutness” in terms of the implication from subcontexts to supercontexts. The weighting of terms in both the user specified elements and the overall contexts is additionally influenced by terms occurring in their sub-components. We will elaborate this further in subsection 3.6.

3.5 Syntax

This subsection details an XPath-like syntax of POLIS. The motivation for this is to provide means to describe the summarisation in the context of structured (e.g. XML) documents. It should be noted that this syntax is only one possible syntax, and other - possibly application-specific - syntaxes could be specified.

A POLIS-expression starts with a root-definition, followed by a specification of the axis of the structured document to be summarised.

```
expression     ::=   | root-element axis-def
root-elem.     ::=   | ’/’
axis-def       ::=   | axis-element
                     | summary-definition
axis-elem.     ::=   | node-def. sum.-def.
                     | node-def ’/’ axis-def
node-def       ::=   | ’*’
                     | ’*’ ’{’ node-restr. ’}’
                     | NAME
                     | NAME ’{’ node-restr. ’}’
node-restr.    ::=   | context-ident.
                     | q.-term
context-id.    ::=   NUMBER
q.-term        ::=   | NAME | NAME q.-term
```

An axis-definition can either be a path-element (called *axis-element* here), or a summary-definition, restricting the number of elements to appear in the summary. Nodes in the axis can be specified explicitly by giving the number of the axis-element to be used, or by specifying query-terms to appear in the node-element.

```
summary-def.       ::=   ’:’ ’{’ sum.-res. ’}’
summary-restr.     ::=   | range-restriction
                         | element-restriction
                         | vague-range-restr.
range-restr.       ::=   ’at_most’ NUMBER
element-restr.     ::=   | NUMBER
                         | NUMBER ’,’ elem.-res.
vague-range-r.     ::=   | ’full’
                         | ’large’
                         | ’medium’
                         | ’small’
                         | ’tiny’
```

The important elements forming the summary can be range-restricted (i.e. allowing a certain number of elements to be returned), element-restricted (i.e. asking for specific elements to be returned) or vaguely range restricted (i.e. not giving an explicit range, but using vague predicates for the range). Vague predicates allow to associate the number of elements used in a summary with a certain predicate. This association could be user-specified, or could be learned from user-provided examples.

NAME ::= $[a-z][a-z\_A-Z0-9]^*$
NUMBER ::= $[0-9]^+$

Finally, a name is defined as any character sequence starting with a lower case letter, followed by any number of letters, digits or underscores. A number is a character sequence consisting of a digit, followed by any number of digits.

3.6 Semantics

Following the specification of a possible syntax for POLIS, we now define the semantics of the summarisation logic. The definition of the semantics will be two staged, starting with a non-probabilistic variant, which is then extended into a full probabilistic semantics. At the moment, no framework exists which can interpret POLIS expressions natively. Instead, POLIS expressions are translated into other target languages (such as probabilistic Datalog, or probabilistic relational algebra [PRA]), for which IR frameworks are available. Having a well defined semantics allows us to assert that translated expressions are functionally equivalent to their POLIS origin. While a formal proof of equivalence is beyond the scope of this paper, figure 2 shows how such a proof could be carried out conceptually. A proof would need to ascertain that both a POLIS expression and its translated Datalog program operate on the same universe of discourse, i.e. both operate on the same textual components. Furthermore, it would be necessary to show that the uncertain reasoning carried out in probabilistic Datalog is equivalent to the probabilities defined in POLIS.

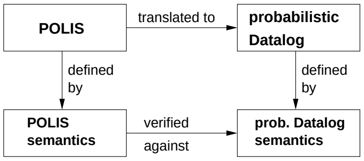


Figure 2: Equivalence of POLIS expressions and their translated Datalog programs can be guaranteed using semantic proofs.

The semantics of POLIS are based on a probabilistic possible worlds interpretation. Traditional possible worlds interpretations postulate the existence of a number of ways “the world could be”, and specify the meaning of logical expressions by defining in which of these worlds the expressions are true. The possible worlds semantics proposed here adds probabilities to the inference process, to model uncertainty of reasoning. For a set of possible worlds, assertions are not just true or false, but have a probability of being true or false.

3.6.1 Interpretation of structured documents

To summarise a structured document, POLIS needs to consider both its content as well as its structure. In the interpretation presented here, contexts in structured documents are seen as agents, which can reason about the possibility of certain worlds (see (Fagin and Halpern, 1994)). The basics of this reasoning we introduced in subsection 3.2.

A structured document is a document that consists of a hierarchy of contexts. In an XML document for example, a context might be seen as the space enclosed by a pair of tags. Terms that occur in contexts are interpreted as propositions, which might or might not be true in the worlds an agent considers possible. In the non-probabilistic case, such propositions will be true in all worlds accessible to an agent (considered possible by an agent). A proposition φ will thus be known to an agent a in world w_i if it is true in all worlds accessible from w_i, i.e. $(M,w_i) \models \varphi \iff \forall w' \in \mathcal{R}_a(w_i) : (M,w') \models \varphi$, where $\mathcal{R}$ is a function that returns for agent a in world w_i all those worlds considered possible by a_i. To allow for more complex knowledge representations than just true propositions, propositions can be either true or false, modelled by $\models_t$ (or just $\models$), and $\models_f$, respectively. The accessibility relation of the interpretation structure contains a relation for each context in the structured document. For the root context of structured documents, we need to introduce an additional, artificial "world", from which the context agent can access possible worlds related with the root context; this world is denoted by "$\circledcirc$". No further assumptions are made about properties of the accessibility relation (i.e. we do not assume it to be reflexive, transitive, symmetric), so that knowledge of contexts is only influenced by the knowledge generalisation rule and the distributed knowledge axiom (Fagin et al., 1995). To illustrate this formal definition, we show for the sample XML document in figure 1 the valuation of the proposition "Bulgaria". The term "Bulgaria" occurs in the second paragraph of the document, the valuation for "Bulgaria" with respect to p_2 is thus *true*. Adopting the closed world assumption implied by a two-valued logic, the proposition is false in all other contexts:

Table 2: Truth values for proposition "Bulgaria" from sample document in an unaugmented interpretation of contexts.

world w'	proposition φ	truth value $\pi(w')(\varphi)$
$w_{\circledcirc}$	Bulgaria	*false*
w_{doc1}	Bulgaria	*false*
w_{body1}	Bulgaria	*false*
w_{text1}	Bulgaria	*false*
w_{p1}	Bulgaria	*false*
w_{p2}	Bulgaria	*true*
w_{p3}	Bulgaria	*false*

In addition to the propositions which are true in a specific context (i.e. knowledge of a context), context knowledge should be augmented by knowledge of its subcontexts. To allow supercontexts in structured documents to have the same knowledge as their subcontexts, or even new knowledge achieved by combining knowledge of subcontexts, we need to define the *augmentation* of supercontexts' knowledge with propositions known to subcontexts. For the interpretation of a structured context c, this means that the interpretation of propositions φ with respect to the augmented c-context c_a would be equivalent to the interpretation of φ wrt a basic, unstructured context b in which all φ occur which are true in subcontexts of c.

Augmentation of contexts for a two-valued logic is somewhat problematic, as terms which do not occur in a context are assumed to be false. We thus present the augmentation for a two-valued logic, however, a more precise and coherent definition of augmentation will be given in subsection 3.6.2. The augmented knowledge of a context is the distributed knowledge of its subcontexts. However, the definition of distributed knowledge given by Halpern *et al.* (Fagin et al., 1995) is too limiting for our purposes, as it does not allow for the combination of knowledge in the case of non-intersecting accessibility relations. To overcome this, we define *united knowledge* of a group of contexts G such that a proposition φ is true in G iff a member of G knows φ to be true, and all other members give evidence to φ being true. More formally:

$$(M,w) \models < G\ \varphi > \iff \exists s \in G : (M,w) \models < /s\ \varphi > \text{ and}$$
$$\forall s \in G : ((M,w) \models < /s\ \varphi >).$$

We will use $\lhd\!/(s_1 \cdots s_n)\rhd \models \varphi$ as a short form of "$G \models \varphi$ where $(s_1 \cdots s_n) \in G$". We can now define the knowledge augmentation of a supercontext in terms of the united knowledge of its subcontexts. A proposition φ is defined to be true in an augmented context if φ is true in the supercontext and the united subcontexts give evidence to true. More formally, this can be expressed as:

$$((M,w) \models < /d_{(s_1 \cdots s_n)}\ \varphi > \iff (M,w) \models < /d\ \varphi > \text{ and}$$
$$\forall w' \in \mathcal{R}_d(w) : ((M,w') \models < /(s_1 \cdots s_n)\ \varphi >.$$

In the case of a two-valued logic, only propositions which are true in all subcontexts can augment the knowledge of a supercontext. For the sample document in figure 1, no term occurs in all three paragraphs; the knowledge of either supercontext w.r.t. to the paragraphs thus remains unaugmented.

While the above definitions cater for a non-probabilistic interpretation of structured documents, the inclusion of probabilities into the model can be achieved by prefixing propositions in contexts with their probabilities of being true or false. This leads to the interpretation of probabilities in probabilistic content as:

$$(M,w) \models < /s\ \rho^t\ \varphi > \iff$$

$$\rho^t = \mu_{s,w}(\{w' \mid w' \in \mathcal{R}_S(w) \wedge (M,w') \models_t \varphi\}),$$

where ρ^t is the probability of proposition φ being true. The interpretation for *false* follows analogously. The truth values for proposition "Bulgaria" in augmented contexts are shown in table 3; probabilities were ommitted for clarity. Note that for the case of non-probabilistic knowledge augmentation, the table would look identical to table 2; in table 3, propositions are true with a certain probability only.

Table 3: Truth values for proposition "Bulgaria" in an augmented context. Probabilities ommitted for clarity.

world w'	proposition φ	$\pi(w')(\varphi)$
$w_{\circledcirc(doc1)}$	Bulgaria	*true*
$w_{doc1_{(body1)}}$	Bulgaria	*true*
$w_{body1_{(text1)}}$	Bulgaria	*true*
$w_{text1_{(p1,p2,p3)}}$	Bulgaria	*true*
w_{p1}	Bulgaria	*false*
w_{p2}	Bulgaria	*true*
w_{p3}	Bulgaria	*false*

Applying the possible worlds scenario, an agent considers a number of worlds possible.Each proposition φ has a truth value in each of these worlds. For each worlds w, w' in $\mathcal{R}_d(w)$, and agent a, a probability $\mu_{a,w}(w')$ is defined. The probability that a proposition φ is true is equal to the sum of the probabilities of the possible worlds in which φ is true (Nilsson, 1986).

3.6.2 Three-valued logic

Information retrieval traditionally adopts an open-world assumption, where propositions for which no valuation is given are assumed to be unknown. To model this explicitly, we extend the definitions of our two-valued logic to a three-valued interpretation, which handles *unknown* as an explicit valuation. Interpretations of propositions thus allow for an additional valuation:

$$\begin{array}{lcl} (M,s) \models_t \varphi & \iff & \pi(s)(\varphi) = \text{true} \\ (M,s) \models_f \varphi & \iff & \pi(s)(\varphi) = \text{false} \\ (M,s) \models_u \varphi & \iff & \pi(s)(\varphi) = \text{unknown} \end{array}$$

Furthermore, both the definitions for united knowledge and for the augmented knowledge of contexts need to be amended to handle the explicit formulation of unspecified knowledge. The definition of united knowledge thus extends such that a proposition φ is true in G iff a member of G knows φ to be true, and all other members give evidence to φ being true or φ being unknown. More formally:

$$(M,w) \models < G\ \varphi > \iff \exists s \in G : (M,w) \models < /s\ \varphi > \text{ and}$$
$$\forall s \in G : ((M,w) \models < /s\ \varphi > \text{ or } (M,w) \models_u < /s\ \varphi >).$$

We will use $\lessdot/(s_1 \cdots s_n)\gtrdot\ \models \varphi$ as a short form of "$G \models \varphi$ where $(s_1 \cdots s_n) \in G$".

The inclusion of *unknown* as a valuation also allows a less strict interpretation of the augmented knowledge of supercontexts. Whereas before knowledge could only be augmented by facts true in all subcontexts, it is now possible to augment a supercontext's knowledge by propositions true only in some subcontexts, as long as all other subcontexts give evidence to unknown only. A proposition φ is defined to be true in an augmented context iff either φ is true in the supercontext and the united subcontexts give evidence to true or unknown or φ is true or unknown in the supercontext and the united subcontexts give evidence for true. More formally, this can be expressed as:

$$\begin{array}{l} ((M,w) \models < /d_{(s_1 \cdots s_n)}\ \varphi > \iff (M,w) \models < /d\ \varphi > \text{ and} \\ \quad \forall w' \in \mathcal{R}_d(w) : ((M,w') \models < /(s_1 \cdots s_n)\ \varphi > \text{ or} \\ \quad\quad (M,w') \models_u < /(s_1 \cdots s_n)\ \varphi >)) \text{ or} \\ \quad (((M,w) \models < /d\ \varphi > \text{ or} \\ \quad\quad (M,w) \models_u < /d\ \varphi >) \text{ and} \\ \quad\quad\quad \forall w' \in \mathcal{R}_d(w) : (M,w') \models < /(s_1 \cdots s_n)\ \varphi >). \end{array}$$

This concludes the extension of our two-valued logic to a three-valued interpretation.

3.6.3 Semantics of POLIS expressions

The meaning of any POLIS expression is geared towards providing a ranking of important parts, and to only consider a limited number of important parts as a substitute for the overall document. With the above definitions for the interpretation of knowledge in contexts, we can say that the probability that an important part should be part of a summary should correspond to the degree that an important part is "about" the document (or context) in which it appears.

This probability of an important part forming part of the summary can then be expressed as the sum over the probability of all terms in a subcontext cooccurring with those respective terms in the augmented supercontext. "Sum" here does not necessarily mean an arithmetic sum, but any combination operation seem fit. Different underlying assumptions about the distribution of terms in the contexts involved would require different combination strategies.

To derive a summary in this model, the probability of propositions in important parts needs to be compared to the probability of propositions in the whole document to be summarised. As the surrounding supercontext in structured documents does not necessarily contain any propositions at all, the overall document context will be augmented with the knowledge of its subcontexts, i.e. with those propositions which occur in all its subparts. This can be stated more formally as:

$$p(\mathrm{ip} \mid \mathcal{S}) := \sum_{t \in \mathrm{ip}_{(s_1 \ldots s_n)}} p(\mathrm{ip}_{(s_1 \ldots s_n)} \mid t) * p(t \mid d_{(s_1 \ldots s_n)})$$

To give a more intuitive example of what these semantics mean, we again revert to the sample document given in figure 1. We applied the POLIS expression **/doc/body/text/p:**{**at_most 3**} to the document, to get the degree to which a paragraph is representative of the overall document. Using the above probabilistic semantics, we arrived at the following probabilities:

"aboutness"	doc. element
0.991764	/doc[1]/body[1]/text[1]/p[1]
0.999955	/doc[1]/body[1]/text[1]/p[2]
0.999822	/doc[1]/body[1]/text[1]/p[3]

According to the derived probabilities, the second paragraph is most representative of the overall document.

4 EVALUATION

4.1 Experimental Setup

The effectiveness of a summarisation approach is usually measured as the ability to correctly identify the most relevant bits of information contained in a text. For evaluating traditional summarisation approaches, test corpora were devised consisting of a number of original texts, and their (human generated) corresponding extracts. The effectiveness of a summarisation approach is measured as the number of sentences cooccurring in both the automatic and manual summaries.

4.1.1 Implementation

To evaluate the summarisation logic presented here, it is necessary to apply the summarisation model to adequate test corpora. Currently, no native interpreter for POLIS expression exists. Instead, logical expressions are translated into probabilistic Datalog programs, which can be executed on existing IR frameworks. Details of this translation process are beyond the scope of this paper, and have been presented elsewhere.

4.1.2 Test Collection

For the summarisation logic presented here, evaluation needs to be performed on a corpus of structured documents, rather than a corpus of unstructured documents. For this reason, we used the AQUAINT document corpus, together with evaluation metrics provided by the Document Understanding Conference (DUC). The set of documents from the AQUAINT corpus used for evaluation consists of a number of newswire texts in English, drawn from three different sources: the Xinhua News Service, the New York Times, and the Associated Press. The corpus is split into fifty subcorpora, such that each of those subcorpora contains 25 documents which cover one common topic. Individual documents in the corpus are tagged using SGML markup, such that structural information of documents is available. The finest level of structural granularity available for the AQUAINT corpus is at paragraph level, which for this corpus coincides with individual sentences.

The task of DUC is to provide for each of the fifty topics a 250 word summary, derived from the documents provided for that topic. To generate a summary, POLIS ranks the structural elements occurring at the user specified granularity by the degree to which they are "about" the topic. The actual summaries are created by producing terms from the ranked elements until the threshold is reached.

To compare the summaries generated using POLIS to those extracts provided as part of the evaluation metrics of DUC, we applied the ROUGE evaluation framework[1]. To evaluate machine generated summaries, ROUGE analyses the generated summary and a set of reference summaries for co-occurrence of n-grams (where *n* ranges from 1 to 4) of words, and from this data derives a measure of overlap between the generated and the reference summaries (Lin and Hovy, 2003). ROUGE was derived from the BLEU framework used in machine translation research (Papineni et al., 2001), however, where BLEU provides a precision oriented measure of overlap, ROUGE employs a recall oriented metric.

4.2 Results

The main focus of our experiments was placed on the effectiveness of POLIS. However, we briefly present efficiency results first, before discussing in detail the quality of POLIS summaries.

[1] http://berouge.com/default.aspx

4.2.1 Efficiency

All our experiments were carried out on an Intel Pentium 4 2.6 GHz machine, with 2GiB of main memory. We timed the processing using the built-in Linux “time” command. All reported times are real time.

Table 4: Performance per 1000 terms

Topic	No. Terms	total Time	Time per *k*-term
D0601A	13225	45.434s	3.435s
D0634G	9428	44.214s	4.689s
D0609I	4616	9.643s	2.089s

To give a feel for the overall performance of our approach, we timed the processing for the smallest (D0609I), a medium (D0634G), and a large topic (D0601A). We measured the size of the topics as the number of terms present in the subcollections after indexing. The results are shown in Table 4. The system performs best for the smallest subcollection, and slows down for the larger topics. However, the speed decrease from the medium to the large topic is lower than the loss in performance going from the small to the medium topic. The time it takes to process 1000 terms actually indicates a performance increase for the large topic compared to the medium one. We believe this performance to be reasonably good, especially considering that summarisation systems rarely need to perform in real time.

4.2.2 Effectiveness

For each topic in the DUC document set, four human generated summaries were provided as part of the evaluation metric. Using ROUGE, co-occurrence statistics were calculated for the POLIS generated summaries and the manual summaries provided. To smoothen out the effect of individual reference summaries in the evaluation process, jackknifing was employed. Precision, recall, and $F_{0.5}$-measure values where reported at 95% confidence level. For comparison, the 2006 DUC average values are provided in the graphs. Note that no DUC2006 performance measures were available for topics 36 – 50. The POLIS summaries achieved precision values generally slightly below DUC average. However, the precision curve follows that of the DUC average, indicating a general correspondence between DUC automated summaries and POLIS summaries. Similarly to precision, recall for POLIS summaries was slightly below the DUC average. However, the recall values were generally better than for precision. The $F_{0.5}$-measure combines both precision and recall values, and confirms that POLIS performance is slightly below DUC average, with general performance following that of the DUC participant summaries.

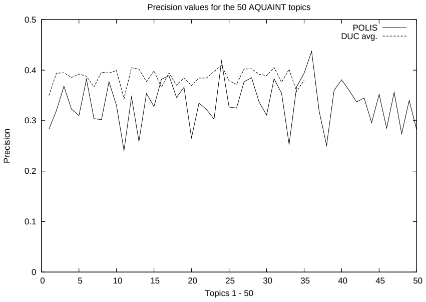


Figure 3: Precision values for the 50 DUC topics

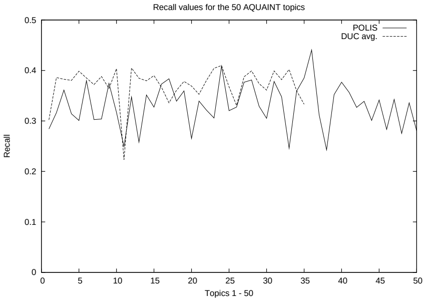


Figure 4: Recall values for the 50 DUC topics

4.3 Discussion

The efficiency achieved with the POLIS generated summaries shows the feasibility of a logic based summarisation approach. The precision of POLIS generated summaries compared to the references provided was 4.962 percentage points below the DUC average at 35 topics, whereas recall values were 3.883 percentage points below DUC average at 35 topics. We believe that the slightly below average performance of POLIS is the result of the current summarisation

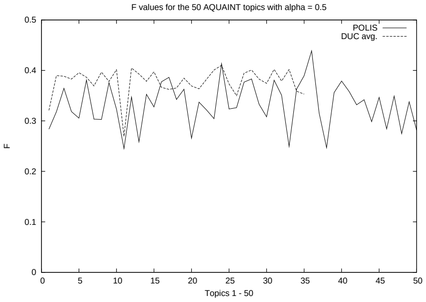


Figure 5: $F_{0.5}$ values for the 50 DUC topics

model applied to a multi-document summarisation corpus. POLIS tries to establish the salience of textual elements as their “aboutness” of the surrounding context. For a single document, for stylistic reasons alone different elements will differ in content. For a multi-document summarisation approach however, different documents reporting on the same topic will feature similar elements, thus having similar scores with respect to the document collection. An improved future summarisation model will thus not only use aboutness as a measure of salience, but will augment this score with a measure of how dissimilar an element is to all other potential summary elements.

5 CONCLUSION

We have presented the syntax and semantics of POLIS, a logic for summarisation of structured documents. We have shown how to use an established semantic structure (Kripke structures) for describing the semantics of POLIS. The main contribution of this paper is to formalise the syntax and semantics of POLIS. We have sketched how formal semantics can be used to prove the correctness of POLIS expressions translated to probabilistic Datalog for processing. Furthermore, we have applied the concepts of POLIS to an established summarisation evaluation corpus, the DUC AQUAINT corpus. Our experiments show that POLIS is able to summarise documents in a reasonably short time, with a summary quality only slightly below that of other, contemporary summarisers. This proves the feasibility of a logic-based approach to summarisation for structured documents. We consider POLIS a novel and complementary contribution to IR where logic-based approaches in the past have been proposed for describing retrieval only.

REFERENCES

Alam, H., Kumar, A., Nakamura, M., Rahman, A. F. R., Tarnikova, Y., and Wilcox, C. (2003). Structured and unstructured document summarization: Design of a commercial summarizer using lexical chains. In *ICDAR*, pages 1147–1152.

Clark, J. and DeRose, S. (1999). XML path language (XPath) 1.0.

Edmundson, H. (1969). New Methods in Automatic Extraction. *JACM*, 16(2):264–285.

Fagin, R. and Halpern, J. (1994). Reasoning About Knowledge and Probability. *JACM*, 41(2):340–367.

Fagin, R., Halpern, J., Moses, Y., and Vardi, M. (1995). *Reasoning about Knowledge*. MIT Press, Cambridge, Massachusetts.

Fuhr, N., Lalmas, M., Malik, S., and Kazai, G., editors (2005). *INEX 2005 Workshop Pre-Proceedings.*

Fum, D., Guida, G., and Tasso, C. (1985). Evaluating importance: A step towards text summarization. In *IJCAI*, pages 840–844.

Hovy, E. and Lin, C. (1997). Automated text summarization in SUMMARIST.

Knight, K. and Marcu, D. (2002). Summarization beyond sentence extraction: a probabilistic approach to sentence compression. *Artif. Intell.*, 139(1):91–107.

Kupiec, J., Pedersen, J., and Chen, F. (1995). A trainable document summarizer. In *SIGIR ’95*, pages 68–73.

Lin, C.-Y. and Hovy, E. (2003). Automatic evaluation of summaries using n-gram co-occurrence statistics. In *NAACL ’03*, pages 71–78.

Litkowski, K. C. (2004). Summarization experiments in duc 2004. In *DUC 2004*.

Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts. *IBM Journal of R&D*, 2(2):159–165.

Meghini, C., Sebastiani, F., Straccia, U., and Thanos, C. (1993). A model of information retrieval based on a terminological logic. In *SIGIR-93*, pages 298–307.

Nilsson, N. J. (1986). Probabilistic logic. *Artif. Intell.*, 28(1):71–88.

Papineni, K., Roukos, S., Ward, T., and Zhu, W. (2001). Bleu: a method for automatic evaluation of machine translation.

Rau, L. F., Jacobs, P. S., and Zernik, U. (1989). Information extraction and text summarization using linguistic knowledge acquisition. *Inf. Process. Manage.*, 25(4):419–428.

Reimer, U. and Hahn, U. (1988). Text condensation as knowledge base abstraction. In *Proc. IEEE-88*, pages 338–344.

Sakai, T. and Sparck-Jones, K. (2001). Generic summaries for indexing in information retrieval. In *SIGIR ’01*, pages 190–198.

Saravanan, M., Raman, S., and Ravindran, B. (2005). A probabilistic approach to multi-document summarization for generating a tiled summary. In *ICCIMA ’05*, pages 167–172.

van Rijsbergen, C. J. (1986). A non-classical logic for information retrieval. *Comput. J.*, 29(6):481–485.

Wolf, C. G., Alpert, S. R., Vergo, J. G., Kozakov, L., and Doganata, Y. (2004). Summarizing technical support documents for search: expert and user studies. *IBM Syst. J.*, 43(3):564–586.