Exploiting relevance assessments for the creation of an experimental test collection to evaluate systems that use annotations

Maristella Agosti, Tullio Coppotelli, Nicola Ferro, and Luca Pretto

Department of Information Engineering, University of Padua, Italy
{agosti, coppotel, ferro, pretto}@dei.unipd.it

Abstract. This study addresses the lack of test collections to evaluate search algorithms that exploit annotations in order to increase the retrieval effectiveness. In particular, it proposes a new method to automatically create a test collection without human effort. The new collection is based on an already existing collection that has to be divisible in at least two disjunct subsets. Those subsets constitute the documents and candidate annotations of the new collection. We make use of the relevance assessments of the original collection to create documents-annotation relationship that best fit the human behavior in annotation creation. The proposed method builds a graph starting from the relevance assessments and, by processing such graph, it finds information which individuate the best links between documents and annotations.

Categories and Subject Descriptors
H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 [**Systems and Software**]: Performance evaluation.

General Terms
Experimentation, Performance, Measurement, Algorithms.

Keywords
Evaluation of an annotation service, construction of test collections.

1 Introduction

In the *Information Retrieval (IR)* field a lot of research is done to identify new and better algorithms which aim at improving the retrieval effectiveness. Such algorithms usually have a twofold goal: to increase the number of relevant documents retrieved and to rank them better.

In this perspective, the annotations made on documents offer an interesting possibility for improving the retrieval performances. Indeed, the additional information contained in the annotations and the hypertext which connects annotations to documents allow us to define search strategies which merge multiple sources of evidence in order to increase the system efficacy. As an example, annotations may be written to expand the content of the original document or to present different points of view on the document itself [1]; an extensive study on annotations is [2]. As a consequence, hidden facets of the documents may be brought to light by annotations and can be exploited to achieve a better match with the user's information needs. With this respect an interesting and promising approach has been proposed in [3].

However, the proposed approach is still lacking a full experimental evaluation mainly because an experimental collection with annotation is missing. Therefore, this paper addresses the problem of setting an adequate experimental test-bed for search algorithms which exploit annotations and discusses a flexible strategy to create test collections with annotated documents; an initial proposal in this direction has been made in [4].

The interesting characteristic of the proposed strategy is that it enables the fast creation of a reliable test collection without direct human intervention. To achieve this goal the main idea is to start from an existing test collection without annotations where documents are objectively divisible in more than one set and to use the relevance assessments available for the existing test collection as a way of introducing annotations into the original collection, thus avoiding the time-consuming manual creation of them.

Moreover this strategy is not limited to the creation of a single collection, rather by using as a starting point collections with different characteristics, it allows the creation of new collections with the same characteristics as the original one (monolingual or multilingual, general or specialized) without introducing subjective bias.

The approach, that is presented in this study, has been fully implemented and the strategy was initially tested using the multilingual *Cross-Language Evaluation Forum (CLEF)*[1] collection. In Section 2 we specify the context in which we operate, in Section 3 we formalize the adopted model. In Section 4 we propose an algorithm that uses relevance assessments to match annotations to documents while in Section 5 and 6 we justify and present an algorithm to weight the pool of documents for each topic and to match new annotations. Conclusions are presented in Section 7.

2 Annotations in Digital Libraries

We all are familiar with written annotations, since it is common to take notes while writing or reading a document, but the annotation is a more complex and multifaceted concept, which ranges from considering annotations as metadata about the annotated object to regarding them as an additional content.

Annotations can be considered as metadata, that is additional data which concern an existing content and clarify the properties and the semantics of the annotated content. With this aim, annotations have to conform to some specifications, which define the structure, the semantics, the syntax, and, maybe, the values that annotations can assume.

On the other hand, annotations can be regarded as an additional content which concerns an existing content, meaning that they increase the existing content by providing an additional layer of elucidation and explanation of it. This viewpoint about annotations embraces what usually occurs to us when we think about the activity of reading a document and adding notes to it: explanation and clarification of words or passages of the document by expounding on it, providing a commentary on it, and finally completing it with personal observations and ideas. In addition, this viewpoint entails an intrinsic dualism between annotation as *content enrichment* and annotation as *stand-alone document*: the former considers annotations as mere additional content regarding an existing document and, as a result, they are not autonomous entities, but in fact rely on previously existing information resources to justify their existence; the latter regards annotations as real documents and autonomous entities that maintain some sort of connection with an existing document.

In addition, annotations are not limited to paper form but we can take full advantage of them by providing a *Digital Library System (DLS)* with annotation capability [1]. The primary effect of introducing annotations is to enrich the *Digital Library (DL)* content; for example, by using annotations the content of a document can be broadened with personal considerations, to propose different points of view or to underline text passages that need further discussion. In addition, annotations allow users to actively integrate DLs in their way of working to create a cooperative environment where annotations become the medium for users to communicate with each other. Moreover, this set of annotations is a new and interesting context of research for information retrieval and it is important to have the instruments for evaluating the algorithms proposed to exploit this context.

Another important characteristic of annotations is their heterogeneity. Annotations in DLs are created by different authors with different backgrounds and at different times: the user who annotates a document may know recent information about the topic which the document author did not know; he or she may disagree with the document content and would like to communicate this different opinion to the readers of the document. This heterogeneity is a key-point that allows a dynamic improvement in the content of the document and by using this new information it is possible to better estimate the relationship between documents and queries, a feature which is so important in document retrieval.

Finally, it is possible to annotate different media such as text, video or images and annotations themselves can be multimedia objects. However, in this study we primarily focus on the use of textual annotations to annotate textual documents.

Annotation Constraint

When we work with annotations, an assumption is needed (which will be called *annotation constraint* in the rest of the paper): each annotation can annotate one and only one document or annotation, that is,

[1] `http://www.clef-campaign.org/`

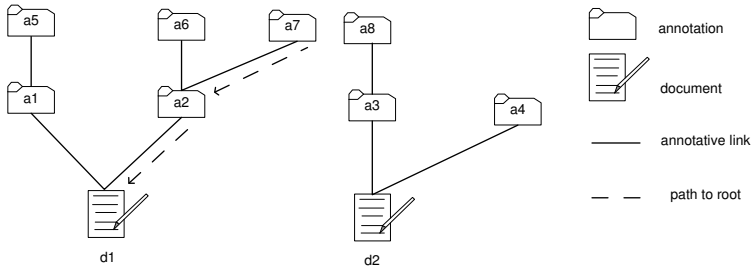

Fig. 1: Annotation constraint.

each annotation is written for exactly one *Digital Object (DO)*. As a consequence of this constraint, the set of documents and annotations become a forest and it is now possible, starting from an annotation, to identify the root document of an annotation thread. Consider, for example, figure 1 where annotation a_7 belongs to the tree rooted in d_1 and note that this would not be possible if a_7 could annotate also a_3 [3].

This annotation constraint does not limit the real behaviors of users who usually write an annotation only after the reading of a single document. It also has the main advantage of allowing the proposed algorithms to assign each annotation, independently of its depth, to a single document.

3 A new approach to collection construction: the model

Usually in IR the creation of an experimental test collection starts from scratch, but the problems which need to be addressed are various [5]: it is necessary to find a suitable set of documents, to manually create annotations and topics and, finally, to evaluate the document relevance to each topic that is for each topic we need to find the documents in the collection that are relevant.

It is important to stress that before proceeding we need to examine the following problems: first, it is not possible to use assessors to create the set of annotations because, to maintain their heterogeneous nature, we need a wide range of annotations written by different authors in different periods of time; second, we actually do not have a sufficient number of experiments to use the pooling method, which has been defined in TREC [5], and which is currently widely adopted. This prevents us from reducing the documents which the human assessor needs to assess for each topic, please note that for an assessor it is not possible to assess the relevance of each document of the collection to a topic. Finally, if we were able to overcome those limitations the standard collection creation process will still be expensive and time consuming.

To deal with these problems we propose a different strategy that involves the use of an already existing test collection as a starting point and the automatic construction of a parallel collection of related annotations.

The proposed new strategy has the following advantages:

1. it reduces the effort of creating the new collection;
2. the results obtained evaluating systems with the new collection are comparable with the previous results obtained on the original test collection; this allows, testing the systems with the original collection and the new one, the direct performance comparison between systems that use annotations and systems that do not use them. This permits to evaluate the improvements, and in general the changes, that new algorithms or their refinements introduce in systems;
3. the approach exploits existing pools to deal with a sufficient number of experiments;
4. the process allows the fast creation of multiple collections with different characteristics that allow the evaluation of the algorithm behavior in different contexts.

We define a test collection as a triple $C = (D, T, J)$ where D is the set of documents, T is the set of topics and J is the set of relevance assessments defined as $J = D \times T \times \{0, n-1\}$ where n are the possible relevance levels of a document to a topic. We choose $n = 2$ (binary relevance) and we use a test collection whose documents are divisible in at last two different sets: $D = D_1 \cup \hat{A}$. We select D_1 as the set of documents in the new collection and $\hat{A}$ as the set of candidate annotations that is $A \subseteq \hat{A}$ where A is the set of actual annotations. In our assumption $D_1 \cap \hat{A} = \varnothing$. Starting from the set of relevance assessments

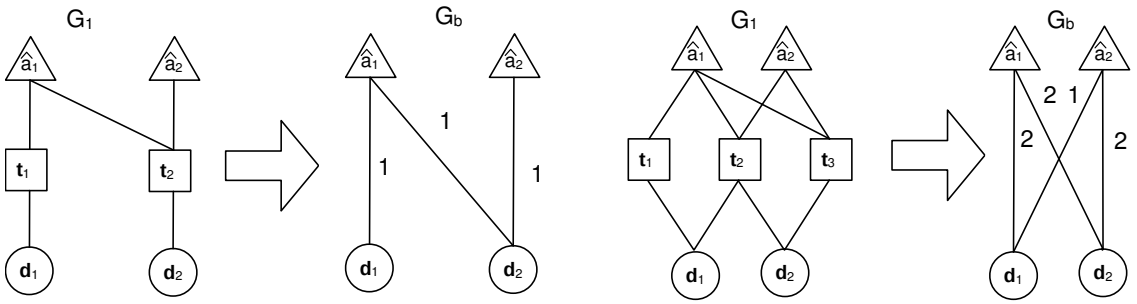

Fig. 2: Examples of the construction of graph G_b, starting from graph G_1.

J we create a graph $G = (V, E)$ where $V = D_1 \cup T \cup \hat{A}$ and $E = (D_1 \cup \hat{A}) \times T$. The graph in undirected and each edge represents a relevance assessment between a document or a candidate annotation and a topic (please remember that documents and annotations are all documents in the original collection). In C all documents in the pool are assessed and we use this information to distinguish edges in $E = E_0 \cup E_1$, being $E_0 \cap E_1 = \varnothing$, where E_1 is the set of assessments judged as relevant by assessors and E_0 is the set of assessments judged as not relevant by assessors.
We define $G_1 = (V, E_1)$ as the graph where all edges represent an human assessment, that is this graph is built linking documents and annotations to topics on the basis of human behavior. Our intuition is that using the information contained in this graph it is possible to identify relationships between annotations and documents that permit us to introduce annotations in the original collection. In Section 4 we exploit G_1 with the aim of finding the best matches between documents and annotations, in Section 5 we exploit the information contained in E_0 with the aim of obtaining more matches.

4 Exploiting the Relevance Assessments to Annotate Documents

Once graph $G_1 = (V, E_1)$ is given, the problem of matching a candidate annotation with a suitable document can be addressed. The matches should respect the annotation constraint that one annotation can annotate only one document. This section describes an algorithm which makes use of the positive relevance assessments to match a candidate annotation with a document. The first aim of the algorithm is to match each candidate annotation with the most suitable document. When more than one match is possible, the algorithm heuristically tends to choose matches which maximize the number of annotated documents—indeed, maximizing the number of annotated documents is the second aim of the algorithm.

The algorithm works in two phases. In the first phase it constructs a weighted bipartite graph G_b on the basis of G_1, i.e. the graph whose edges represent positive relevance assessments. In the second phase the algorithm works on the weighted bipartite graph G_b to properly match a candidate annotation with a document.

The construction of the weighted bipartite graph $G_b = (V_b, E_b)$ is immediate: the vertices of G_b are all the vertices of G_1 which represent documents or candidate annotations, that is $V_b = D_1 \cup \hat{A}$, and an edge between candidate annotation $\hat{a}$ and document d exists if and only if $\hat{a}$ and d have been judged relevant to at least one common topic, that is $t \in T$ exists such that edges $\hat{a}$-t and t-d are in E_1. Moreover, a weight is assigned to each edge $\hat{a}$-d in E_b, which gives the number of common topics between $\hat{a}$ and d. These weights take account of the fact that when $\hat{a}$ and d are assessed as relevant to more than one common topic at the same time, it is reasonable to suppose that the bond between the candidate annotation $\hat{a}$ and the document d will be strengthened. In Figure 2 simple examples of the construction of G_b, starting from G_1, are given.

Once G_b is constructed, the algorithm works only on G_b to properly match a candidate annotation with a document. It is this second phase of the algorithm that has the two aims described above. The first aim is that of matching the best possible annotation with a document: this is done considering first the edges with the highest weight. The second aim is that of trying to annotate the maximum number of documents, once the best possible annotations have been considered.

The first aim is achieved by first analysing only the edges with the maximum weight and using all of them to match candidate annotations with their suitable documents. After all the edges with the maximum weight have been analysed, only the edges of immediately lower weight are analysed and so

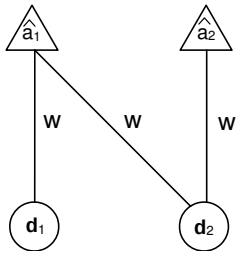

Fig. 3: Example of a conflict. Note that all edges have the same weight w.

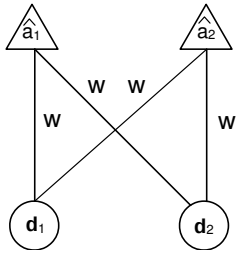

Fig. 4: Example of a deadlock. Note that all edges have the same weight w.

on, until all the edges with a positive weight have been analysed. In other words, the algorithm considers each different layer of edge weight separately—the higher the layer, the higher the quality of the matches. When a layer with a certain weight is considered, only edges with that specific weight are analysed.

The second aim, i.e. trying to annotate the maximum number of documents, is achieved by the conceptual application, layer by layer, of two operators, $O_{\text{conflicts}}$ and O_{random}. The first operator is applied to match a candidate annotation with a document, and also has the task of resolving conflicts like those in Figure 3, where if $\hat{a}_1$ were matched with d_2 it would no longer be possible to annotate document d_1, while the best choice is to match $\hat{a}_1$ with d_1 and $\hat{a}_2$ to d_2. To avoid these conflicts operator $O_{\text{conflicts}}$ first selects all the couples $\hat{a}$-d for which $\hat{a}$ can annotate only one document, like the couple $\hat{a}_2$-d_2 in Figure 3. Then $O_{\text{conflicts}}$ matches candidate annotations with documents in order to annotate the maximum number of documents: for instance, in the case of Figure 3, $\hat{a}_1$ will be matched with d_1, since d_2 has already been annotated. Once an edge $\hat{a}$-d is used in a match, it is marked with the negative weight -1, and all the other edges which are incident with the same candidate annotation $\hat{a}$ are deleted from the graph and no longer considered. $O_{\text{conflicts}}$ is iterated until it resolves all possible conflicts. However, in some cases $O_{\text{conflicts}}$ cannot find a match, since no preferable match is suggested by the topology of the graph. This occurs, for instance, when a kind of deadlock exists (see Figure 4). In this case an operator O_{random} is applied, which randomly selects one of the possible matches between a candidate annotation and a document. As usual, when a match, that is an edge $\hat{a}$-d, is selected, that edge is marked with the negative weight -1, and all the other edges which are incident with $\hat{a}$ are deleted. The algorithm applies iteratively $O_{\text{conflicts}}$ and O_{random} operators until all the edges with the weight under consideration have been examined. Then a lower weight is examined and so on, until all *positive* weights have been examined.

Finally, edges marked with the negative weight -1 give the desired matches of candidate annotations with documents.

In figure 5 one possible solution of a deadlock problem is proposed. There are four equiprobable edges and $O_{conflicts}$ cannot match any annotation to document then O_{random} is applied and delete one edge with probability 0.25. In the example, after the deletion of edge $d_1 - \hat{a}_2$, it is possible to annotate d_2 with annotation $\hat{a}_2$. In the next execution step is reapplied $O_{conflicts}$ that now can match $\hat{a}_1$ with d_1, and not $\hat{a}_1$ with d_2 because d_2 respect to d_1 is already annotated. Please, note that applying O_{random} it is no more possible to find an unique solution to the matching problem but this is not relevant respect to our aim to find the maximum number of matches.

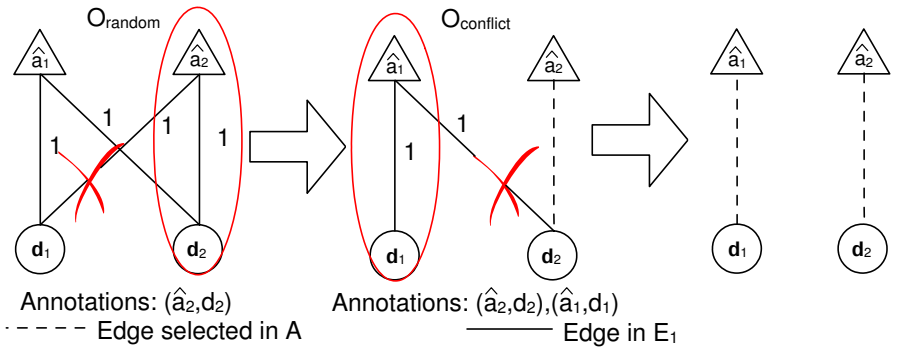

Fig. 5: Example of application of O operators.

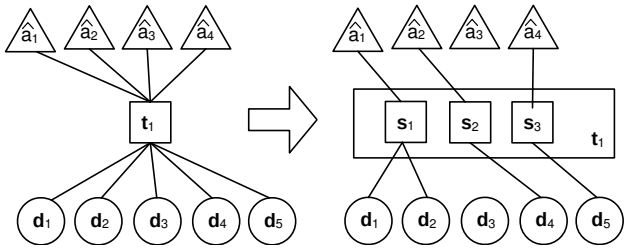

Fig. 6: Subtopic creation.

5 Subtopic view of the graph

In Section 4 we have defined the method for matching a certain number of annotations against documents using the relevance assessments of the assessors. In this way it is possible to match only the annotations that are assessed relevant to, at least, one topic; it is important to note that we cannot decide anything about annotations that are still in the pool but are not relevant to any topic, because, for construction, in graph G_1 they are not connected to any topic. In this Section we state that in the original collection there still exists a certain number of good couples document-annotation that are not matchable with the previous method and we present a practical justification for their existence. The idea is to build the graph using not only the assessments of assessors but the whole pool and the content of documents and annotations as well.

We define A_2 as the set of vertex â identified with the previous algorithm, this is the set of annotations incident to negative edges, and E_2 as the edges incident to A_2, then we define $G_3 = G(V/A_2, E/E_2)$ where G_3 is the graph obtained using the whole pool for each topic in the original collection and removing, due to the annotation constraint, all the candidate annotations already matched. In this new graph we have, for each topic, a set of documents and annotations that are no longer judged relevant to the topic, since those relevant were already assigned, but that still present some bounds between them. Considering the graph that represents all documents of G_3 inserted in the pool for a single topic, we have found that it is possible to group them in subsets on the basis of their content. For each subset we ideally create a new topic that we call subtopic $S_1, S_2, \ldots, S_k$ and now we are not interested in the relevance of documents or annotations to the original topic, since we know by construction, that is not possible, but we are interested in the matches between documents and annotations relevant to the same subtopic like $\hat{a}_1, d_1$ and d_2 in figure 6.

To better understand the concept of subtopic we propose the following example from the CLEF collection: the original topic T is about "Alberto Tomba's skiing victories" but the IR system used to create the pool also introduced in the pool non-relevant documents like those about skiing competitions where Alberto Tomba does not participates (S_1), those where he participates without winning (S_2) or documents about the social life of that "very important person (VIP)" (S_3). We aim to match documents $d_i \in D, \hat{a}_j \in \hat{A}$ where both d_i and $\hat{a}_j$ are incident to some subtopic S_k. It is important to stress that we are not really interested in identifying those subtopics but their existence is used as a practical justification for the matches that we search using the whole pool. Clearly, it is not trivial to find documents and annotations relevant to the same subtopic, without knowing which one they are, and hence we cannot

use the previous algorithm that relies on human assessments because we no longer have their support; we need to find a new automatic strategy to evaluate the quality of the relationship between candidate annotations and annotated documents. Focusing on this goal, on the basis of the graph and the document content, we introduce four automatically calculated parameters and we use those parameters to compute a unique weight on the graph G_3. Finally we try to match the greatest number of good quality annotations against the greatest number of documents; please note that the problem differs from the one of the previous graph G_b because now we cannot assign all the annotations due to the very poor quality of some of them. Although we obtained good experimental results, the choice of parameters and their weight is not definitive and has to be tuned. The proposed algorithm works with the weighted graph independently of the way the weight is computed. Section 6 introduces the new parameters and the adopted method.

6 Exploiting the whole pool to annotate the graph

The four independent parameters, that we propose to use, are automatically calculated from the graph, they are:

1. affinity;
2. a score obtained using an information retrieval tool;
3. generality; and
4. temporal nearness.

The affinity P_a informs us about the superimposition in the content of two or more topics such that increasing the affinity also increases the similarity of topics involved. This parameter assumes a numerical value that ranges between 0 and 1, as for the other three parameters, and it is computed as follows. Starting from two topics T_i and T_j we define $D_i, D_h \subseteq D$ and $\hat{A}_i, \hat{A}_j \subseteq \hat{A}$ where all documents in $D_i, D_j, \hat{A}_i, \hat{A}_j$ are in the pool for topic T_i and T_j.
$T_t = \{(d_i, \hat{a}_j) \subseteq D \times \hat{A} | d_i \in D_i \cap D_j \text{ and } a_i \in \hat{A}_i \cap \hat{A}_j\}$. The affinity is the cardinality of T_t normalized with the maximum value of T_t.
The efficacy of this parameter is linked to the increase in the probability that two documents contain similar content if they belong to topics with increasing affinity. Another important application of affinity is the elimination from the graph of documents belonging to topics with very low affinity because those documents probably do not belong to any subtopic. Experimental results have shown a sort of transitivity in the affinity of topics, with score degradation, which is not actually supported by theoretical results.

We have decided to use an IR tool to evaluate the strength of every match in the graph. In the creation of the pool we have lost the information about the score and the order with which the systems have ranked each document, so it is not odd to use an Information Retrieval system to rank the result of the work of other IR systems. Moreover, we are interested in the evaluation of the relevance of a document to an annotation and we are no longer interested in the relevance of the document to a topic so we do not violate the relevance assessments of assessors. We create an index from all the candidate annotations and we query the system using the content of each document. In this way we obtain an ordered list of possible annotations, we choice the first K results and we intersect them with the content of the graph obtaining two results: 1) all edges between the document and the annotations that do not belong to the list are erased from the graph; 2) all the remaining edges in the graph are weighted with the score P_{ir} assigned by the system. Please note that intersecting the results with the graph we delete those couples that have not a path in G_3, while applying 1 we delete from the graph the couples that have worst matches. In this phase we are only weighting the edges in the graph, and deletions are performed assigning to deleted edges the weight corresponding to the maximum cost.

In contrast to what happened with the graph G_1 we have found that when using the whole pool it is no longer true that increasing the number of topics in witch a couple belongs also increases the quality of the couple. In this case the generality of the annotation increases , because an annotation included in a lot of topic pools necessarily has to be a generic one. Hence we compute the generality score P_g based on the inverse of the number of edges incidents to the annotation's vertex (that is the number of topics per annotation).

The last parameter to be introduced is P_t, that is the temporal nearness. Our choice was to increase the score of matches between documents that are temporally close beside their order because we found,

as happened to affinity, that the probability of finding a match of good quality increases with an increase in the temporal nearness of the two documents.

To evaluate the strength of each document-annotation couple it is convenient to compute a unique score based on the parameters introduced. This formula is computed with the intersection of the result of each previous parameter and applying the following score computation:

Score $S = a_A * P_A + a_{IR} * P_{IR} + a_G * P_G + a_T * P_T$ with $a_A + a_{IR} + a_G + a_T = 1$.

Moreover, this formula has the advantage of being open for new parameters and operators.

Once an unique weight S is computed, there still exists a trade off between the number of documents that can be annotated and the quality of those annotations. If we select all possible annotations, ignoring their weight, the result is a collection with poor matches while if we select only the best matches we run the risk of annotating few documents and with an unequal distribution. The algorithm proposed in [4] resolves this trade off by proceeding in phases. By setting the parameters of the algorithm it is possible to choose the best fitting mixture of good quality annotation and annotated documents.

7 Conclusions

The experimental results obtained confirm the quality of the collection created and the possibility of automatically create a collection of adequate dimensions without reassessing the pool of documents and, since the annotations are obtained by simulating human behavior, the process improves the collection reliability.

In future work we plan to study the affinity parameter and its applications in different contexts. We would investigate the possibility of applying the graph theory to the graph with the double aim of reducing the number of documents that assessor need to assess and post-validating their assessments. The method proposed establishes a starting point for this study.

Acknowledgements

The work reported in this paper has been partially supported by the DELOS Network of Excellence on Digital Libraries, as part of the Information Society Technologies (IST) Program of the European Commission (Contract G038-507618), in particular in the context of Task 4.10 of Work Package 4 on User Interfaces and Visualization and Task 7.4 of Work Package 7 on Evaluation.

References

1. Agosti, M., Ferro, N.: Annotations: Enriching a Digital Library. In Koch, T., Sølvberg, I.T., eds.: Proc. 7th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2003), Lecture Notes in Computer Science (LNCS) 2769, Springer, Heidelberg, Germany (2003) 88–100
2. Agosti, M., Bonfiglio-Dosio, G., Ferro, N.: A Historical and Contemporary Study on Annotations to Derive Key Features for Systems Design. International Journal on Digital Libraries ((in print))
3. Agosti, M., Ferro, N.: Annotations as Context for Searching Documents. In Crestani, F., Ruthven, I., eds.: Proc. 5th International Conference on Conceptions of Library and Information Science – Context: nature, impact and role, Lecture Notes in Computer Science (LNCS) 3507, Springer, Heidelberg, Germany (2005) 155–170
4. Coppotelli, T.: Creazione di una collezione sperimentale per la valutazione di sistemi di reperimento dell'informazione che utilizzino le annotazioni (in Italian). Master's thesis, Department of Information Engineering, University of Padua (2006)
5. Harman, D.K., Voorhess, E.M.: The Text REtrieval Conference. In Harman, D.K., Voorhess, E.M., eds.: TREC. Experiment and Evaluation in Information Retrieval. MIT Press, Cambridge (MA), USA (2005) 3–19