Question Answering Summarization of Multiple Biomedical Documents

Zhongmin Shi, Gabor Melli, Yang Wang, Yudong Liu, Baohua Gu,
Mehdi M. Kashani, Anoop Sarkar and Fred Popowich

School of Computing Science, Simon Fraser University
Burnaby, BC V5A 1S6, Canada

Abstract. In this paper we introduce a system that automatically summarizes multiple biomedical documents relevant to a question. The system extracts biomedical and general concepts by utilizing concept-level knowledge from domain-specific and domain-independent sources. Semantic role labeling, semantic subgraph-based sentence selection and automatic post-editing are involved in the process of finding the information need. Due to the absence of expert-written summaries of biomedical documents, we propose an approximate evaluation by taking MEDLINE abstracts as expert-written summaries. Evaluation results indicate that our system does help in answering questions and the automatically generated summaries are comparable to abstracts of biomedical articles, as evaluated using the ROUGE measure.

1 Introduction

With the rapid development of biological and medical research in the last decade, the volume of biomedical scientific articles has greatly increased. For instance, over 2,000 new articles are being added to the MEDLINE database every day. It is extremely difficult for physicians and researchers in medicine and biology to build up their own knowledge base from existing publications and update it daily. Therefore automatic methods such as summarization and question answering (QA) that can quickly understand and find the main points of biomedical articles are becoming more essential.

Domain-independent summarizers, such as WebSumm [1], Newsblaster[1] and Alias-I[2], have been used to generate summaries of biomedical articles [2], However, when tuning a summarizer to a particular domain, domain specific information would improve the quality of summaries. Gaizauskas et al. proposed TRESTLE (Text Retrieval Extraction and Summarization Technologies for Large Enterprises) that relies on named entity annotations and scenario templates to generate single sentence summaries of pharmaceutical news archives [3]. Centrifuser [4] is a summarization system that computes topical similarities among documents using a tree structure-based calculation and extracts sentences from

[1] http://www.cs.columbia.edu/nlp/newsblaster/.
[2] http://www.alias-i.com/.

topic-relevant documents. Elhadad and McKeown introduced a summarizer that generates a patient-specific summary from journal medical articles [5]. In this system documents are first categorized into main clinical tasks, from which a set of templates are built and matched with patient records. Patient-relevant templates are then merged and ordered to generate fluent English text.

In this paper we introduce BioSquash, a question-oriented extractive summarization system on biomedical multi-documents that are relevant to a question. The system was based upon a general-purpose summarizer, Squash [6]. We purpose a method to utilize concept-level characteristics from a domain-specific ontology, UMLS (Unified Medical Language System[3]), and a domain-independent lexicial reference system, WordNet[4]. Details of the system design are described in the rest of this paper as follows. §2 provides a high-level description of the BioSquash system. The automatic annotation of the input documents to be summarized is described in §3 and §4. Construction of the semantic graph and the sentence extraction step based upon concept-level characteristics are discussed in §5. §6 introduces our redundancy elimination and sentence ordering strategies to produce more readable summaries. The evaluation on experimental results is given in §7. Some discussions and future work are brought forward in §8.

2 The System Architecture

The BioSquash system has four main components: the Annotator, Concept Similarity, Extractor and Editor modules, as illustrated in Fig. 1. The system starts off by annotating the documents and the question text with syntactic and shallow semantic information in the **Annotator module**. These annotations do not provide sufficient semantic background when we study the relations among concepts in documents and questions. The actual semantic meanings of both general and biomedical concepts as well as the ontological relations among these concepts are obtained in the **Concept Similarity Module**.

The annotations and conceptual information are then fed to two summarization stages: the first is the **Extractor module**, which focuses on content selection and aims to optimize the ROUGE[5] score; while the next stage, the **Editor module**, focuses on linguistic readability. In the Extractor module, a semantic graph is constructed based on the semantic role labeling and the conceptual information of documents as well as the question text. Sentence selection is performed by sub-graph selection on the semantic graph. Sentence redundancy is also measured and used to create sentence clusters related to the topic question. The Editor module orders sentences from the sentence clusters provided by the Extractor, eliminates irrelevant content from long sentences and finally produces the summary conforming to the length limit.

[3] http://www.nlm.nih.gov/research/umls/.
[4] http://wordnet.princeton.edu/.
[5] Recall-Oriented Understudy for Gisting Evaluation, http://haydn.isi.edu/ROUGE/

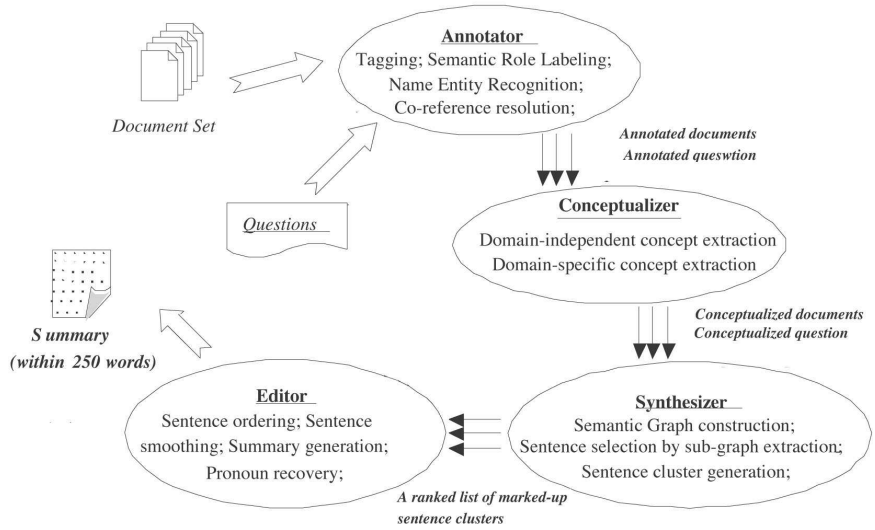

Fig. 1. The overall system design of BioSquash

3 The Annotator Module

The annotations used in the BioSquash system include output of a statistical parser [7], a named-entity recognizer and a semantic role labeler. The named-entity recognition (NER) sub-module categorizes atomic text elements into predefined named entity classes. We use Alias-I's Lingpipe system to identify persons, organizations, locations, numeric entities, pronoun entities as well as biomedical entities as defined in the GENIA ontology[6].

A semantic role is the relationship that a syntactic constituent has with a predicate. Typical semantic roles for arguments of the predicate include Agent, Patient, Instrument, and semantic roles for adjuncts include indicating Locative, Temporal, Manner, Cause, among others. The task of semantic role labeling is, for each predicate in a sentence, to identify all constituents that fill a semantic role and to determine their roles, if any [8, 9]. Recognizing and labeling semantic arguments is a key task for answering "Wh-" and other more general types of questions in the summarization task. Automatic semantic role labeling methods have been discussed in depth in [8]. The semantic role labeling (SRL) for documents and questions is produced by transducing the output of the statistical parser using our own SRL system [10], which is trained on the semantic annotations provided by the CoNLL-2005 data-set, a modified version of the annotation provided by the Penn PropBank data-set [11]. The following example illustrates the input and output of the SRL sub-module:

[6] http://www-tsujii.is.s.u-tokyo.ac.jp/genia/topics/Corpus/genia-ontology.html.

Example 1
Input: (S (NP (NP (DT The) (NN processing)) (PP (IN of) (NP (NNP LcnA)))) (VP (VBZ involves) (NP (NNP LcnC))) (. .))

Output: (S ($_{A0}$ (NP (NP (DT The) (NN processing)) (PP (IN of) (NP (NNP LcnA))))) (VP (VBZ $_{PREDICATE}$ involves) ($_{A1}$ (NP (NNP LcnC)))) (. .)) [7]

4 The Concept Similarity Module

Identification of similarity between questions and sentences in documents is crucial to our QA summarization task, in which sentences in documents are selected based on their similarities with other sentences in documents and questions. Specifically, the sentence similarity metric is used to choose the significant "group" of sentences and to decide the relevance of a sentence to the question. A sentence similarity confined to word surface patterns and simple string matching would however fail in cases of:

- **identifying synonyms**. Synonyms with different lexical forms are not taken into account in the sentence similarity metric using only the word-based approach. This problem is especially sensitive in biomedical documents, since many biological and medical substance names have various lexical forms. For instance, when the question asks" the role of the gene BARD1 in the process of BRCA1 regulation", we would expect terms like "function"/"character", "BARD 1", "Breast Cancer 1 Protein" to be considered as similar.
- **identifying hypernym-hyponym relations**. If the question contains hypernyms of words in the sentence, the word-based approach does not consider the sentence to be similar to the question. For instance, the occurrence of the words "arbovirus", "bacteriophage" and "viroid" in the sentence should improve the sentence significance when the question involves their hypernym, "virus".
- **word sense disambiguation**. The word-based approach would take lexically identical words as the same, even though they occur different senses in a particular context.

We define **Super Concept**, as a synonym, hypernym (*is-a*) or holonym (*part-of*) of a concept. Therefore, super concept is reflexive and transitive: 1) Any concept is a super concept of itself; 2) If concept A is a super concept of B and B is a super concept of C, then A is a super concept of C. We say two concepts are related ontologically if one is the super concept of the other.

The BioSquash system recognizes each concept by a **Concept ID** (CID), a unique identification of each distinct entity, event and relation. Concepts with the same CID are synonyms. In addition, hypernyms of each concept are also provided, therefore the system would ideally not have above three problems after the conceptualization.

[7] We use the Role Set defined in the *PropBank Frames scheme* [9].

Concepts and their hypernyms are extracted from two public ontologies: WordNet for domain-independent concepts and UMLS for domain-specific concepts. We apply a CPAN module[8], WordNet::SenseRelate::AllWords, to select the correct sense of each word. The module is an implementation of a word sense disambiguation algorithm that measures similarity and relatedness based on the context of the word and the word sense glosses in WordNet [12]. Table 1 shows an example of sentence annotation that can allow matching according to concept similarity by matching CIDs.

Table 1. An example of a sentence annotated with concept ID's to allow matching according to concept similarity

WORD	The	processing	of	LacZ	involves	LcnC	.
WordNet_CID	-	(13366961)	-	-	(02602586)	-	-
UMLS_CID	-	-	-	(C0022959)	-	(C1448241)	-

5 The Extractor Module

The Extractor takes the results from the Annotator and Concept Similarity Modules (see §3 & §4) and provides relevant sentences to the Editor module (see §6). The extractor module performs the following tasks.

5.1 Concept Identification

The first task of the Extractor is to locate the concepts that exist in the document set. Given the deep syntactic and shallow semantic annotation, a relatively complete set of concepts is extracted. Three types of concepts are located: ontological concepts, named entities and noun phrases. An ontological concept is a phrase that Concept Similarity Module has connected to one of the available ontologies (WordNet or UMLS). For example, the well-studied organism *Pseudomonas aeuroginosa*, the phrase *nucleotide sequence*, and the word *located* would likely be assigned a concept because the Concept Similarity Module would recognize it as a member of UMLS. Similarly, a named entity concept is a text phrase that Annotator has identified to be a certain named entity type. For example, the recently discovered protein *macrophage inflammatory protein-1a* could be identified by a named-entity recognizer (NER) but not by Concept Similarity Module because the ontology is dated and/or curated. Finally, a word that is not recognized as either a named entity or a member of the available ontologies, but is recognized to be a noun phrases by the statistical parser, is also labelled as a concept.

[8] http://www.cpan.org/.

5.2 Text Graph Creation

Once the concepts have been identified within the documents, the next task performed by the Extractor is to identify linguistic and semantic relationships between them. The relationships between concepts result in graph edges. More specifically, the semantic relations of a document can be given by the semantic labeler. As in [13] a *semantic graph* is used as a semantic representation of the document. Because the system works on multiple documents for one question, the semantic graph represents the semantic relations for all the documents by sharing the common nodes and links. An example of a small portion of the semantic graph constructed from a set of documents is shown in Figure 2. Constructed from the output of the Annotator, the text graph contains the information in the text of the documents and question that is essential to topic-based multi-document summarization.

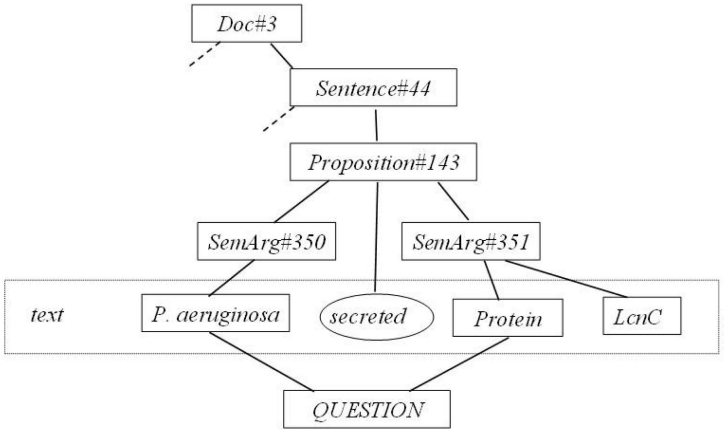

Fig. 2. Subgraph of the overall text graph associated with the proposition *P. aeruginosa secreted the protein LcnC*. The dashed lines indicate multiple edges that likely exist but are not included in the figure. In this example the question asks for information about proteins associated with P. aeruginosa.

5.3 Concept and Proposition Significance

Given a text graph, each concept is assigned a **significance** score and then each proposition is scored based on concepts it contains. The value of this score is based on 1) the number of edges to questions, documents and propositions; 2) whether the type is an ontological concept, named entity or noun phrase. Next, each proposition in every sentence from every document is given a significance

score, which will be used to rank the relevance of each proposition to the summary. The contribution to the significance score is calculated as the summation of the individual significance values for each unique entity in the sentence. Further details about assigning the significance score are described in [6].

5.4 Covering the Concept Space

The selection of propositions occurs sequentially, and a fixed number of sentences is returned. This number is large enough to result in more sentences than strictly needed for the summary length limit. The selection of the first proposition is simply based on the largest significance score, with ties broken randomly. Once the first proposition is selected, similar propositions are of less value to Extractor. These similar propositions are penalized in order to ensure that other interesting topics were selected. This process is iterated until the required number of sentences was selected. Further details about the penalty function are introduced in [6].

6 The Editor Module

The task of the Editor module is to produce a fluent summary. To achieve this, we order all the sentences based on their significance scores produced by the Extractor module, and select the highest scoring subset of the sentences as the candidate sentences for the summary. Those sentences are then re-ordered by a 2-phase sentence ordering algorithm and a summary candidate is generated after compressing the re-ordered sentences. The sentence compression step deletes words or phrases that involve the use of discourse or chronological markers that can affect fluency of the summary after re-ordering.

We propose a two-phase ordering algorithm to assign an **Importance** score to each sentence and order them. In the first phase, the importance score g of each sentence s_i is computed as a linear combination of a list of features: $g(s_i) = w_1F_1 + \ldots + w_nF_n$, where F_j is the value of the jth feature equal to either 0 or 1. w_j is the corresponding interpolation weight of F_j. We manually set w_j based on a study of existing summaries. The following features are used to calculate the importance score: information importance from Extractor module, question and sentence overlap, first and last sentences in the document and sentence-length cutoff.

The sentences are then re-ordered by their importance scores. In order to choose the sentences that are coherent to their neighbors in the summary, we calculate the similarity score of two sentences based on their Longest Common Subsequences (LCS), as the second phase of ordering. More specifically, to choose the kth sentence m_k of the summary, each of the rest sentences is measured against the sum of two scores in said ordering phases: $m_k = argmax_{s \in S'}\ g(s) + LCS(s, m_{k-1})$, where S' denotes the set of sentences that have not been selected in the summary. Note that, although we have not experimented with different sentence similarity measures, LCS may be replaced by other techniques, for instance, edit distance and n-gram comparison.

7 Experiments and Evaluations

7.1 Data

One of the challenges to developing a document summarizer for biomedical documents is measuring the quality of the machine-generated summaries. The ideal scenario is to have the summaries evaluated by experts in biomedicine. Unfortunately, as in most research settings, we did not have access to a pool of experts to evaluate our summaries. The more typical method of evaluation is to use an n-gram based algorithm to compare the machine-generated summaries to a set of human written summaries (for which we also need domain experts to produce multiple summaries for the same question/topic). The annual Document Understanding Conference (DUC)[9] for example hires humans to write summaries of newspaper articles that answer a specific questions, such as: *What countries have chronic potable water shortages and why?* To the best of our knowledge however, no dataset has been explicitly created to test automated summarization from the biomedicine domain.

We solve this problem by transforming data from the Ad-hoc Retrieval task of Genomics Track at the 2005 Text Retrieval Conference (TREC)[10] in order to test the BioSquash system. The original TREC task was: given a question on relationships that exist among biological substances and/or processes, retrieve a set of documents that are relevant to the question from a 4.5-million document subset of the MEDLINE database. To transform this dataset to our needs we simply treat the paper abstracts as substitutes for expert-written summaries. Our assumption is that a paper abstract selected to be relevant by a human judge will closely approximate the summarized answer sought from the question. Figure 3 contains a sample question from the TREC task.

From the TREC data we selected the 18 questions (the precise question IDs are provided in Fig. 4) with the most number of documents associated with it. The questions with a large number of documents were required in order to effectively apply ROUGE, a recall oriented n-gram matching measure of summarization quality [15]. For each TREC question the relevant abstracts are separated into three subsets: 5 abstracts as peers(each peer abstract is taken to be a human written summary and compared our machine generated summary), 30 abstracts that will be summarized by BioSquash and 30 abstracts to be used as references by ROUGE.

A fourth set of abstracts was associated to each of the 18 selected questions. This set is based on the documents retrieved by one of the Information Retrieval (IR) systems that participated in the TREC task [14][11]. This set allows us to present results for a real-life system that both retrieves appropriate documents and produces a summary from them. Figure 3 presents one of selected questions

[9] http://www-nlpir.nist.gov/projects/duc/

[10] http://ir.ohsu.edu/genomics/.

[11] this system's performance was within the average of all submitted systems (Means of Average Precision: 0.1834).

Question: *Provide information about the role of the gene Apolipoprotein E (ApoE) in the disease Alzheimer's Disease.*
Summary: *The raised frequency of the epsilon 4 allele in the patients with onset Alzheimer's disease was of a lower magnitude than that in United States and Canadian studies. The frequency of E4 alleles was increased (chi 2 = 42; df = 1; p < 10) among patients with Alzheimer's disease compared with a Danish control population. Apolipoprotein E is associated with Alzheimer's disease neurofibrillary tangles and beta-amyloid protein in senile plaques. The relative epsilon 4 allele frequency was 0.472 in LBD, 0.513 in AD-CVD, 0.405 in presenile AD, 0.364 in senile AD, and 0.079 in vascular dementia. To clarify the association of ApoE polymorphism with Alzheimer's disease and vascular dementia in Japan, 13 patients with early onset sporadic Alzheimer's disease, 40 patients with late onset sporadic Alzheimer's disease, 19 patients with vascular dementia, and 49 non-demented control subjects were analysed. Apolipoprotein E sigma4 allele is associated with Alzheimer's disease in familial and sporadic cases, but the associations of ApoE sigma4 allele and vascular dementia and/or ischemic cerebrovascular disease are still controversial. Alzheimer's disease is associated with an increased frequency of the apolipoprotein E type epsilon 4 allele. The epsilon 4 allele has also been shown to reduce the age at onset of dementia in AD in a dose dependent manner, with the epsilon 2 allele having an opposing effect. Apolipoprotein E epsilon 4 allele frequency among Alzheimer's disease patients is increased compared to control subjects and is influenced by the presence of other genetic factors and age at symptom onset.*

Fig. 3. One of the TREC questions ($q117$) and the corresponding summary generated by BioSquash from documents retrieved by the IR system [14].

and the corresponding summary generated by BioSquash from documents retrieved by the chosen IR system.

In summary, the following sets of abstracts are available for each of the 18 questions:

- **Peer Abstracts**: Five of the human selected abstracts are treated as independently written human expert summaries.
- **Relevant Abstracts**: 30 of the human selected abstracts are used to create a summary with BioSquash. This summary is supports the scenario in which documents are manually selected by a human expert and the task is to automatically summarize the documents.
- **Retrieved Abstracts**: 30 of the system retrieved abstracts are used to create a summary with BioSquash. This summary supports the scenario in which the human expert only provides the question and both the retrieval and summarization are automatically performed.
- **Reference Abstracts**: Up to 50 of the human selected abstracts are used as the full set of gold standard summaries when computing the ROUGE score for the peer "summary" or machine generated summary.

7.2 Evaluation Measures and Results

We follow the evaluation methodology of DUC 2005 and DUC 2006 that used ROUGE-2 and ROUGE-SU4 to measure performance. ROUGE-n is an n-gram recall-oriented measure to compare a candidate summary with a set of reference summaries [15]. ROUGE-SU4 is similar to ROUGE-n except that it involves bi-grams with maximum skip distance of 4 [15].

As described in §7.1, our experiment's ROUGE scores measure three different sets of candidate summaries (peer abstracts, summaries of relevant abstracts and summaries of retrieved abstracts) against the reference abstracts. The evaluation of summaries based on retrieved abstracts is related to the setting which an IR module is used to retrieve several documents relevant to the query, and then our system is used to produce a question-focused summarization. Table 2 shows ROUGE-2 and -SU4 scores for the three sets, among which summaries of relevant abstracts outperform others on both ROUGE-2 and -SU4. The results indicate that our question-focused machine generated summaries contain information that is more pertinent to the question when compared to human-written abstracts (which were not written with any particular question in mind, but rather summarize the document for which the abstract was written). Figure 4 illustrates ROUGE-2 and -SU4 scores of the three sets of candidate summaries for all chosen questions.

Table 2. ROUGE-2 and -SU4 scores of the three sets of candidate summaries. For peer abstracts, scores listed are the average of ROUGE scores of 5 peer abstracts.

Candidate Summary	ROUGE-2	ROUGE-SU4
Summaries of relevant abstracts	**0.0697**	**0.1300**
Summaries of retrieved abstracts	0.0669	0.1248
Peer abstracts (average)	0.0690	0.1118

8 Conclusion and future work

Text summarization in a professional domain, e.g., biology and medicine, is a very challenging task due to the need of domain knowledge in understanding domain-specific contents and a great amount of co-operation from domain experts, especially with respect to evaluation of summarization systems. In this paper we proposed a QA-based summarization system, BioSquash, that automatically produces a summary of biomedical multi-documents relevant to a question. The system utilizes conceptual information extracted from both domain-independent and domain-specific ontologies to create fluent 250-word summaries relevant to a user question.

Due to the absence of expert-written biomedical summaries, in the evaluation phase we instead treat MEDLINE abstracts as expert-written summaries

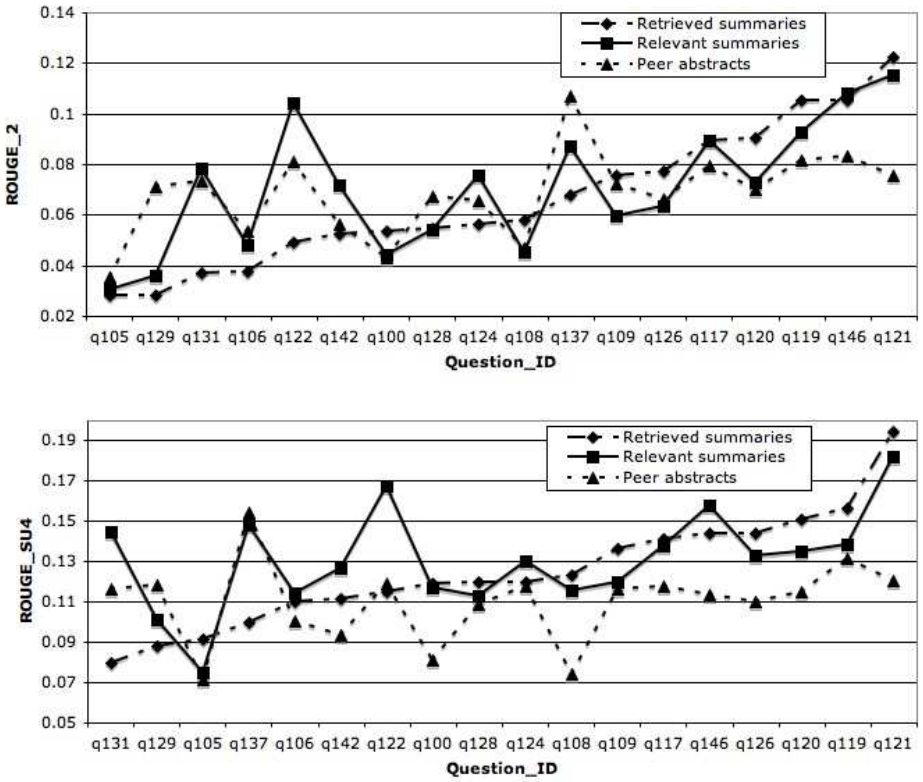

Fig. 4. ROUGE-2 (upper) and -SU4 (lower) of three candidate summary sets for all selected questions, sorted by the performance of BioSquash on abstracts that were retrived by the IR system.

when calculating ROUGE scores. We do not know how close this approximate evaluation would be to the ideal evaluation procedure before a certain amount of expert-written summaries is available. However, our experimental results indicate that: 1) machine-generated summaries of abstracts known to be relevant to the question have better quality in terms of n-gram matching than the human written abstracts that are oblivious to the question; 2) machine-generated summaries of abstracts that were retrieved using an IR system with the question as the query are comparable to abstracts known to be relevant to the question.

The evaluation involves no linguistic readability of summaries, since the work needs significant amount of expert participation that is not available to our system at this stage. We are planning a more comprehensive evaluation on the BioSquash system by collecting human sources, e.g., expert-written summaries and domain expert assessors in future work. We plan to evaluate our system on full articles in addition to abstracts.

References

1. Mani, I., Bloedorn, E.: Summarizing similarities and differences among related documents. Information Retrieval **1**(1) (1999) 1–23
2. Damianos, L., Day, D., Hirschman, L., Kozierok, R., Mardis, S., McEntee, T., McHenry, C., Miller, K., Ponte, J., Reeder, F., van Guilder, L., Wellner, B., Wilson, G., Wohlever, S.: Real users, real data, real problems: the mitap system for monitoring bio events. In: Proceedings of BTR2002, The University of New Mexico (March 2004)
3. Gaizauskas, R., Herring, P., Oakes, M., Beaulieu, M., Willett, P., Fowkes, H., Jonsson, A.: Intelligent access to text: integrating information extraction technology into text browsers. In: Proceedings of HLT 2001, San Diego (2001) 189–193
4. Kan, M., McKeown, K., Klavans, J.: Domain-specific informative and indicative summarization for information retrieval. In: Workshop on text summarization (DUC 2001), New Orleans (2001)
5. Elhadad, N., McKeown, K.: Towards generating patient specific summaries of medical articles. In: Proceedings of automatic summarization workshop (NAACL 2001), Pittsburgh, PA, USA (2001)
6. Melli, G., Wang, Y., Liu, Y., Kashani, M., Shi, Z., Gu, B., Sarkar, A., Popowich, F.: Description of squash, the sfu question answering summary handler for the duc-2005 summarization task. In: Proceeding of DUC-2005, Vancouver, Canada (October 2005) 103–110
7. Charniak, E.: A maximum-entropy-inspired parser. In: Meeting of the North American Chapter of the ACL. (2000) 132–139
8. Gildea, D., Jurafsky, D.: Automatic labeling of semantic roles. Computational Linguistics **28**(3) (2002) 245–288
9. Palmer, M., Gildea, D., Kingsbury, P.: The proposition bank: An annotated corpus of semantic roles. Computational Linguistics **31**(1) (2005)
10. Liu, Y., Sarkar, A.: Using ltag-based features for semantic role labeling. In: Proceedings of the Eighth Workshop on Tree Adjoining Grammars and Related Formalisms: TAG+8, Poster Track, Sydney, Australia (July 2006)
11. Kingsbury, P., Palmer, M.: From treebank to propbank. In: In Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002). (2002)
12. Pedersen, T., Banerjee, S., Patwardhan, S.: Maximizing semantic relatedness to perform word sense disambiguation. In: University of Minnesota Supercomputing Institute Research Report. (March 2005)
13. Mani, I., Bloedorn, E.: Multi-document summarization by graph search and matching. In: Proceedings of the 14th National Conference on Artificial Intelligence, Providence, Rhode Island (1997) 622–628
14. Shi, Z., Gu, B., Popowich, F., Sarkar, A.: Synonym-based query expansion and boosting-based re-ranking: A two-phase approach for genomic information rretrieval. In: the Fourteenth Text REtrieval Conference (TREC 2005), NIST, Gaithersburg, MD. (October 2005)
15. Lin, C.Y., Hovy, E.H.: Automatic evaluation of summaries using n-gram co-occurrence statistics. In: Proceedings of Language Technology Conference (HLT-NAACL 2003), Edmonton, Canada (May 2003)