Concept Frequency Distribution in Biomedical Text Summarization

Lawrence H. Reeve[1], Hyoil Han[1],
Saya V. Nagori[2], Jonathan C. Yang[2], Tamara A. Schwimmer[2], Ari D. Brooks[2]

[1]Drexel University, College of Information Science and Technology, Philadelphia, PA USA
[2]Drexel University, College of Medicine, Philadelphia, PA USA
lhr24@drexel.edu
hhan@ischool.drexel.edu
{svn23, jcy26, tas42} @drexel.edu
ari.brooks@DrexelMed.edu

ABSTRACT
Text summarization is a data reduction process. The use of text summarization enables users to reduce the amount of text that must be read while still assimilating the core information. The data reduction offered by text summarization is particularly useful in the biomedical domain, where physicians must continuously find clinical trial study information to incorporate into their patient treatment efforts. Such efforts are often hampered by the high-volume of publications. Our contribution is two-fold: 1) to propose the frequency of *domain concepts* as a method to identify important sentences within a full-text; and 2) propose a novel frequency distribution model and algorithm for identifying important sentences based on term or concept frequency distribution. An evaluation of several existing summarization systems using biomedical texts is presented in order to determine a performance baseline. For domain concept comparison, a recent high-performing frequency-based algorithm using terms is adapted to use concepts and evaluated using both terms and concepts. It is shown that the use of concepts performs closely with the use of terms for sentence selection. Our proposed frequency distribution model and algorithm outperforms a state-of-the-art approach.

Categories and Subject Descriptors
I.2.7 [**Natural Language Processing**]: Language Parsing and Understanding, Text analysis.

General Terms
Algorithms, Measurement, Performance, Experimentation.

Keywords
Text summarization, concept frequency, biomedicine.

1. INTRODUCTION
Text summarization is a data reduction process. The use of text summarization allows a user to get a sense of the content of a full-text, or to know its information content, without reading all sentences within the full-text. The reduction in the amount of data has the advantage of increasing scale by 1) allowing users to find relevant full-text sources more quickly, and 2) assimilating only essential information from many texts with reduced effort.

There are two different approaches to generating summaries from text: extractive and abstractive [1]. The *extractive* approach extracts sentences or parts of sentences verbatim from text, and is the most common way to perform summarization. The second and substantially more difficult approach is called *abstractive*, and involves generating summary text using natural language processing techniques. Our approach and evaluation uses the extractive approach. A set of identified sentences is used to form a final summary. The task of sentence selection can be considered an information retrieval task, where the set of all sentences within a text are evaluated (scored), and the highest scoring sentences are selected as being the most relevant to a user.

The data reduction offered by text summarization is particularly useful in the biomedical domain. The research presented here is motivated by the task of generating extractive text summaries useful to practicing oncologists, who must continuously find clinical trial study information related to their specialty, evaluate the study for its strength, and then possibly incorporate the new study information into their patient treatment efforts [2], [3]. The U.S. National Institutes of Health Clinical Trials database contains information on over 13,500 clinical trials [4]. In addition, treatment information may be found in databases such as PUBMED, which contains in excess of 12 million citations from over 4,800 journals [5]. These two sources alone make it impossible for a single physician to review every text and assimilate the information contained in them.

The contributions of this work are: 1) to propose the frequency of domain-specific concepts as a feature for identifying salient

sentences in biomedical texts; 2) the development of a new frequency distribution model and a corresponding algorithm which outperforms a state-of-the-art approach; and 3) the use of full-text biomedical sources rather than abstracts. We evaluate several existing, publicly-available summarization systems to determine a performance baseline with biomedical texts using existing approaches. We then evaluate two summarizers using both terms and concepts as unit items to show the use of concepts performs as well as or better than terms.

The paper is organized as follows. Section 2 provides background on text summarization using item frequency as a scoring feature. Section 3 presents a new model and algorithm using frequency distribution to score sentences. Section 4 describes an evaluation of both existing summarization systems as well as recent algorithms using both term and concept frequency as a feature for sentence selection. Section 5 discusses the results of the evaluation. Section 6 provides concluding remarks and suggests areas for future work.

2. BACKGROUND

2.1 Need for Biomedical Text Summarization

Clinical trial studies and other scientific publications usually supply a summary of the paper in the form of an abstract produced by the author(s) of a study. We have identified at least five reasons for wanting to generate text summaries from a full-text source even in the presence of the author's abstract. 1) There exists no 'ideal' summary. An ideal summary is dependent on each user, including factors such as information need and domain background. An author's abstract is one view of an ideal summary, but users may want alternative summaries. 2) The abstract may be missing content from the full-text [6]. 3) Customized summaries can be useful in question-answering systems where they provide personalized information. 4) The use of automatic or semi-automatic summary generation by commercial abstract services may allow them to scale the number of published texts they can evaluate. 5) The generation and evaluation of summaries allows for evaluation of sentence selection methods that may be useful for use in multi-document summarization. The idea is that if sentence selection methods do not work well for single-document summarization, it is unlikely they will identify important data across multiple documents.

2.2 Biomedical Domain Concepts

One way to provide meaning to biomedical documents is by creating ontologies, and then linking information within each document to specifications contained in the ontology using a markup language [7]. Ontologies are conceptualizations of a domain that typically are represented using domain vocabulary [8]. Automatic semantic annotation is the process of mapping instance data to an ontology [9] [10]. The resulting annotations from the semantic annotation processing are what provide the link between information stored within a document and the ontology [7]. In our work, the annotations are then used to identify important areas of a text useful for generating a text summary. In the biomedical domain, the National Library of Medicine (http://www.nlm.nih.gov/) provides resources for identifying concepts and their relationships under the framework of the Unified Medical Language System (UMLS) [11]. UMLS contains many sub-components, but we use only two: Metathesaurus and MetaMap Transfer.

Table 1. A UMLS concept and its concept instances

Concept Name	Concept Instances
Multiple Myeloma	Multiple Myeloma
	Myeloma
	Plasma Cell Myeloma
	Myelomatosis
	Plasmacytic myeloma

The UMLS Metathesaurus contains concepts and real-world instances of the concepts, including a concept name and its synonyms, lexical variants, and translations [12]. The Metathesaurus is derived from over 100 different vocabulary sources. Table 1 shows the example concept "Multiple Myeloma" taken from the Metathesaurus, and displays several of the concept instances associated with the concept. The instances are derived from the vocabulary sources. The key idea is that a single concept may have multiple ways of being expressed (instances). The Metathesaurus organizes the concept instances. The MetaMap Transfer (MMTx) application [13] maps biomedical text to concepts stored in the Metathesaurus as follows. The text-to-concept mapping in the MMTx application is done through a natural language processing approach. Sentences are first identified, and then noun phrases are extracted from each sentence. MMTx proceeds through several stages to map a noun phrase to one or more concepts. Term variants of the phrase are generated, candidate concepts are generated, and a scoring process is done for each candidate concept. The highest scoring concept is then selected as the concept for the phrase. It is possible a noun phrase can map to more than one concept. In this case, no disambiguation step is performed, and MMTx returns multiple concepts. Figure 1 shows an example of MMTx mapping of the phrase "protein kinase CK2". The output shows the phrase, the concept candidates preceded by their score ("Meta Candidates"), and the final mapping of the phrase ("Meta Mapping"). There are six candidate mappings, shown in descending score order. The final mapping takes the highest scoring Meta Candidate (1000). In cases where a phrase cannot be successfully disambiguated, it is possible for MMTx to generate a final mapping consisting of more than one concept.

```
Phrase: "protein kinase CK2."
Meta Candidates (6)
  1000 protein kinase CK2 (casein kinase II) [Amino Acid, Peptide, or Protein,Enzyme]
   901 PROTEIN KINASE [Amino Acid, Peptide, or Protein,Enzyme]
   827 Kinase (Phosphotransferases) [Amino Acid, Peptide, or Protein,Enzyme]
   827 Protein (Proteins) [Amino Acid, Peptide, or Protein,Biologically Active S
ubstance]
   827 Protein NOS (Protein measurement) [Laboratory Procedure]
   827 CK2 [Laboratory Procedure]
Meta Mapping (1000)
  1000 protein kinase CK2 (casein kinase II) [Amino Acid, Peptide, or Protein,Enzyme]
```

Figure 1. MetaMap Transfer mapping of the phrase "protein kinase CK2."

2.3 Frequency as an Extraction Feature

Term frequency was first used in extractive text summarization in the late 1950's [14]. A follow-up study of an analysis of five term frequency methods showed high agreement in sentence selection among the methods [15]. Subsequent research using frequency methods focused on the use of frequency as one feature among many for identifying important sentences, such as cue phrases [16] [17]. Summarization using larger units of text has also been researched. The LAKE system uses keyphrases for summarization [18]. The SUMMARIST system [19] uses WordNet [20] concept counting not for identifying salient sentences, but for topic interpretation. In topic interpretation, concept frequency counting is used to find a node in the concept hierarchy which sufficiently generalizes more specific concepts (e.g., {pear, apple} → fruit). The SUMMARIST authors cite the lack of domain-specific resources as a serious drawback to this approach. Our work uses domain-specific resources exclusively, but we have not used these resources for topic interpretation, only with sentence identification. Most recently, the SumBasic algorithm uses term frequency as part of a context-sensitive approach to identifying important sentences while reducing information redundancy [21]. The use of frequency as a feature in locating important areas of a text has been proven useful in the literature [14] [15] [16] [17]. This is most likely due to reiteration, where authors state important information in several different ways, in order to reinforce main points [22].

2.4 Unit Items for Counting Frequencies

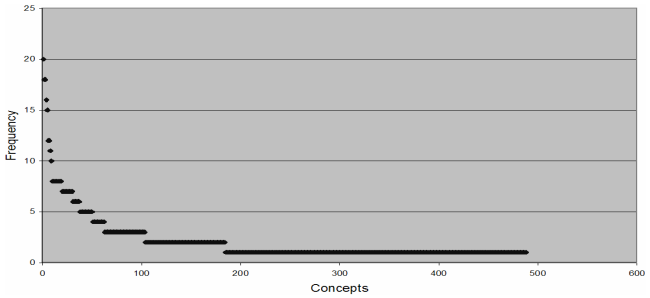
Frequency-based summarization approaches count the appearance of items within the text, and then use the item counts to identify data that has been repeated within a text, which is presumed to be important because it appears multiple times. We call the unit to be counted a *unit item*. A unit item is frequently a term, but can also be another unit, such as a phrase or a concept. Our work focuses on the use of concepts as the unit items. In the evaluation phase described in Section 4, the unit items are concepts as well as terms (words excluding stop words) for the summarizers we implement. For publicly available summarizers in the evaluation, the term unit item is a word.

3. FREQUENCY DISTRIBUTION MODEL

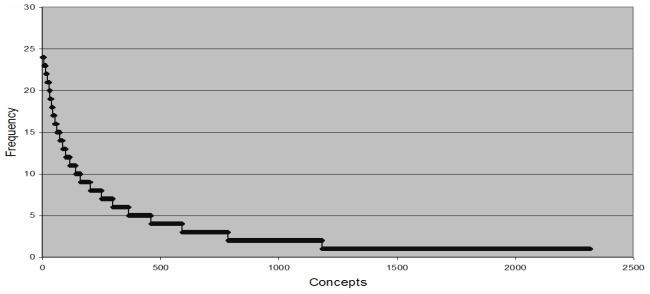
Extractive approaches to text summarization usually follow a model of scoring sentences based on a set of features. The highest scoring sentences are then extracted to form a summary. When using frequency as the only feature, unit items are counted and then each sentence is given a score based on the frequency count of each unit item in the sentence. A key problem in generating summaries is reducing redundancy. Each new sentence in the summary should add new information rather than repeating already included information. Using the highest frequency terms will likely result in the same information repeatedly being selected, with the chance that some additional information is included. In the SumBasic [21] frequency approach, a probability distribution model is first generated, and as each term is used to select sentences, the term probabilities are reduced so that lower probability terms have a better chance of selecting sentences with new information content. This approach is called context sensitivity. This is also related to the idea of finding Maximal Marginal Relevance (MMR), where marginal relevance is defined as finding relevant sentences which contain minimal similarity to previously selected sentences [23].

In this paper, we present a context sensitive approach to scoring sentences based on a frequency distribution model rather than a probability distribution model. The rationale of our approach is that the frequency distribution of terms or concepts ought to appear in the generated summary as closely as possible to the source text. That is, the frequency distribution models of the source and its summary should be as similar as possible.

It is well known that terms in a text follows a Zipf distribution [24]. UMLS resources allow for working at the level of domain-specific concepts rather than terms. In order to use concepts within a frequency distribution model we first show that concepts within a biomedical text also follow a Zipfian distribution. To do this, we first used a corpus of biomedical full-text sources and extracted concepts from abstracts and their corresponding full-text using MetaMap Transfer. The corpus used includes 24 biomedical papers and is described in section 4.1. We used the paper abstracts as an ideal summary, and then compared the distribution models of concepts in the abstract vs. concepts in the full-text. Figure 2 shows the two frequency distribution models. Figure 2(a) shows the distribution of 488 discovered concepts across 24 paper abstracts, while Figure 2(b) shows 2,317 discovered concepts across 24 full-text papers corresponding to the 24 abstracts. As can be seen, both distributions can be characterized as Zipfian distributions. With the observation that both a version of an ideal summary and its corresponding full-text have the same frequency distribution form, we propose an algorithm to generate a summary based on the frequency distribution of the unit items (i.e., terms or concepts) within a full-text.



(a)



(b)

Figure 2. Biomedical text concept distribution across 24 papers. (a) Distribution of 488 discovered biomedical concepts within the paper abstracts. (b) Distribution of 2,317 discovered biomedical concepts within the full-text of the papers.

Figure 3 shows an outline of our algorithm ("FreqDist") to generate a summary given the full-text of some source (source text) using a frequency distribution approach. There are two stages: Initialization and Summary Generation. In the initialization stage, the unit items (terms, concepts, etc.) of the source text are counted to form a frequency distribution model of the text, and a pool of sentences from the source text is created. A summary frequency distribution model is created from the unit items found in the source text, and their frequency counts are initialized to zero. In the Summary Generation stage, new sentences are selected to be added to the summary. Identifying the next sentence to be added to the summary is accomplished by finding the sentence which most closely aligns the frequency distribution of the summary to the frequency distribution of the original source text. For each sentence in the sentence pool, a candidate summary is first initialized to the summary generated so far, and then the sentence is added to the candidate summary. The candidate summary frequency distribution is then compared for similarity to the original source text frequency distribution. This similarity score is assigned to the sentence. After all sentences from the sentence pool have been evaluated for their contribution to the candidate summary, the highest scoring sentence is added to the summary and removed from the sentence pool. This process is iterative, and repeats until the desired length of the summary is reached.

```
Initialization:
// Note: '-model' means  'frequency distribution model'
INITIALIZE source-model to unit-items in source-text;
INITIALIZE summary-model,
                candidate-model from source-model;
              set all frequency values of both models to 0;
INITIALIZE sentence-pool to source-text sentences;

Summary Generation:
REPEAT
  INITIALIZE sentence-pool scores to 0;
  INITIALIZE best-score to 0;
  INITIALIZE best-sentence to first sentence in pool;
  INITIALIZE summary-output to empty sentence list;

  FOR each sentence-entry in sentence-pool
    INITIALIZE candidate-model from summary-model;
    ADD sentence unit-item frequencies to candidate-model;
    SET sentence-entry.score =
            similarity(source-model, candidate-model);

    IF sentence-entry.score > best-score
      SET best-score to sentence-entry.score;
      SET best-sentence to sentence-entry;
    ENDIF
  ENDFOR

  ADD unit-items from best- sentence to summary-model;
  ADD best-sentence to summary-output;
  REMOVE best-sentence from sentence-pool;
UNTIL desired summary size reached or
            sentence-pool exhausted;
RETURN summary-output as a final summary;
```

Figure 3: FreqDist: an algorithm for generating summaries using a frequency distribution approach.

We compared five similarity functions to find which type of function worked best to evaluate a candidate summary's frequency distribution to the original source text frequency distribution. Each frequency distribution (candidate summary and original source text) is modeled as a vector of unit items. Similarity functions are then applied to the two vectors. Figure 4 shows the five similarity functions used. The notations are as follows: *ui* is unit item; *srcUIs* and *sryUIs* are all unit items in source text or candidate summary, respectively; *src(ui)* and *sry(ui)* are indexed unit item in the source text or candidate summary, respectively. Cosine similarity [25], Dice's coefficient [26], Euclidean distance and vector subtraction [27] are all well-known vector comparison methods. In addition, an approach to vector model comparison considering only unit item frequency was tried [28]. Cosine similarity uses the cosine angle value between the vectors for similarity. Dice's coefficient looks at the number of common terms between the two vectors. Euclidean distance measures the distance between the vectors in Euclidean space. For vector subtraction, the absolute value of the difference of each unit item in each vector is summed to form a distance score. The unit item frequency approach attempts to simulate cosine similarity without the computational complexity by only considering unit item frequency [28].

$$score = \frac{\sum_{ui=1}^{srcUIs} sry(ui) \times src(ui)}{\sqrt{\sum_{ui=1}^{srcUIs} sry(ui)^2 \times \sum_{ui=1}^{srcUIs} src(ui)^2}}$$

(a) Cosine similarity

$$score = \left| \frac{2 * count(srcUIs \cap sryUIs)}{count(srcUIs) \cup count(sryUIs)} \right|$$

(b) Dice's coefficient

$$score = \left(\sum_{ui=1}^{srcUIs} (sry(ui) - src(ui))^2 \right)^{1/2}$$

(c) Euclidean distance

$$score = \sum_{ui=1}^{srcUIs} \left| (src(ui) \times sry(ui)) \right|$$

(d) Unit item frequency

$$score = \sum_{ui=1}^{srcUIs} \left| (src(ui) - sry(ui)) \right|$$

(e) Vector subtraction

Figure 4: Similarity functions to evaluate a candidate summary's frequency distribution to the original source text frequency distribution**:** (a) cosine similarity, (b) Dice's coefficient, (c) Euclidean distance (d) unit item frequency, and e) vector subtraction. Notations used: *ui* is unit item; *srcUIs* and *sryUIs* are all unit items in source text and candidate summary, respectively; *src(ui)* and *sry(ui)* are indexed unit item in the source text or candidate summary, respectively.

4. EVALUATION

The purpose of the evaluation is to 1) evaluate the usefulness of concept frequency as a sole feature for identifying salient sentences for extractive text summarization, and 2) evaluate our proposed frequency distribution algorithm "FreqDist" described in Section 3. The evaluation was done by first asking three domain experts to manually generate extractive summaries from 24 biomedical texts (see Section 4.1). A series of automated summarizers (in section 4.5) then generated summaries of the biomedical texts. The output of each summarizer is automatically compared using an automated tool called ROUGE [29] (see Section 4.3). ROUGE generates several scores for each summary. The results are detailed in Section 5. The rest of this section gives details on the evaluation implementation.

4.1 Corpus

A corpus of 24 biomedical texts was generated from a citation database of oncology clinical trial papers. The database contains approximately 1,200 papers physicians feel are important to the field [2]. Of the 1,200 papers cited, 24 were randomly selected. The PDF versions of these papers were then obtained and converted to plain-text format. The papers were manually processed to remove graphics, tables, figures, captions, citation references, and the bibliography section. The resulting text was further split into an abstract text and a full-text source text (without the abstract). The number of papers chosen (24) was based on the minimum requirements of the ROUGE summary evaluation tool [30] as well as the resources available to complete the manual processing of each paper.

4.2 Concept Annotation

Our domain is biomedical text, specifically oncology clinical trial result papers. The Unified Medical Language System (UMLS) Metathesaurus [12] is used as the semantic resource. Concept annotation of each paper is performed using the UMLS MetaMap Transfer tool [13] to perform text-to-concept mapping, as described in Section 2.2. When concepts are used in summary generation, it takes place in two stages: 1) biomedical concept annotation of the source text, and 2) summary generation from the concept-annotated text using the discovered concepts.

4.3 ROUGE

The ROUGE (Recall-Oriented Understudy for Gisting Evaluation) tool (version 1.5.5) [31] developed by the Information Science Institute at the University of Southern California was used. ROUGE is an automated tool which compares a generated summary from an automated system with one or more ideal summaries. The ideal summaries are called models. ROUGE uses N-grams to determine the overlap between a summary and the models. An N-gram can be considered as 1 or more consecutive words. ROUGE was used in the 2004 and 2005 Document Understanding Conferences (DUC) [32] as the evaluation tool. We used the following parameters from the DUC 2005 conference:

-n 2 -x -m -2 4 -u -c 95 -r 1000 -f A -p 0.5 -t 0 -d

Two recall scores are extracted from the output of ROUGE to measure each summarizer: ROUGE-2 and ROUGE-SU4. ROUGE-2 evaluates bigram co-occurrence while ROUGE-SU4 evaluates "skip bigrams" with a maximum distance of 4 words. ROUGE-2 and ROUGE-SU4 are also the measures used by DUC 2005. The recall scores indicate the N-gram overlap between the source text and the model summaries. It is difficult to compare ROUGE results outside of the corpus and model summaries used in the evaluation. For this reason, we gathered several summarizers from publicly-available sources in order to provide some meaningful comparison among them using the same corpus and set of model summaries.

4.4 Model Summaries

To compare summaries generated automatically from systems, we used four models (i.e., four ideal summaries) for each of the 24 papers. The models represent different versions of ideal summaries. The first model is the abstract of the paper (author's summary). In addition, three models from three different domain experts were generated. The domain experts are medical students in their final year. Each was given the task of performing extractive text summarization by selecting 20% of the sentences within a paper which formed the best summary for that paper.

4.5 Summarizers used for evaluation

In this evaluation, six extractive summarizers are used. The BaseLine, FreqDist, and SumBasic summarizers were implemented for this evaluation, and each have multiple variations. The MEAD, Microsoft Word, and SWESUM summarizers are publicly available, and were randomly selected based on their availability. MEAD and SWESUM are research prototypes, while the AutoSummarize feature in Microsoft Word is a commercial application. Each summarizer generated a summary that was equal to 20% of the length of the source text. For example, if a source text consists of 100 sentences, then 20 sentences are selected by the summarizer and presented as the summary. Selecting a summary size was problematic. The news summarization domain typically selects a size of less than five sentences. This represents about 20% of the size of a typical news story [33]. It has been generally thought that a summary should be no shorter than 15% and no longer than 35% of the source text [34]. The following is a brief description of the approaches used by each summarizer.

4.5.1.1 BaseLine

The purpose of the baseline summarizers is to give some indication of the level of performance of a naïve summarization implementation. Two baseline summarizers were implemented. The first baseline summarizer is called LEAD, and it sequentially selects the first 20% of sentences in the source text. The second baseline summarizer is called RANDOM, and it randomly selects 20% of the sentences in the source text.

4.5.1.2 FreqDist

Our FreqDist summarizer implements the algorithm described in Section 3. It can be used to select terms or concepts as the unit to perform frequency analysis on. There are five variations of the FreqDist summarizer. Each variation implements the same FreqDist algorithm in Figure 3, but uses a different vector similarity algorithm in Figure 4 to determine the similarity of unit item frequency distributions of the source text and candidate summaries. When terms were used as unit items, a stop list was applied so that words having low information content (such as 'for') were removed. For the implementation using concepts, the UMLS Metathesaurus was used as the domain-specific resource.

4.5.1.3 MEAD

MEAD [35] is a single- and multiple-document summarizer using multiple features to score sentences. Some of the features include position of sentence within the text, overlap of sentence with the first sentence, sentence length, and a centroid method based on a cluster of related documents. For the evaluation, we used the MEAD Demo located at http://tangra.si.umich.edu/clair/md/demo.cgi. No domain specific knowledge sources were provided to the summarizer.

4.5.1.4 AutoSummarize

The AutoSummarize is a feature of the Microsoft Word [36] word processing software. AutoSummarize is based on a word frequency algorithm. Each sentence in a document is given a score based on the words the sentence contains. Although the exact details of the algorithm are not documented, the online help for the product states that sentences using frequently-used words are given a higher score than sentences containing low frequency words. No domain specific knowledge sources were provided to the summarizer.

4.5.1.5 SumBasic

The SumBasic algorithm [21] is a recent frequency-based algorithm. The original algorithm works using terms. For this evaluation, we have modified it so that the unit items can be terms or concepts. SumBasic incorporates a component for ensuring coverage of weaker concepts within a text. There are four steps in the algorithm. The first is to determine the probability distribution of all concepts found within a source text by computing the number of times a unit item appears in the text divided it by the total number of unit items found in the text. The second step is to score each sentence by summing the probabilities of all unit items within a sentence. The third step determines the sentence to be extracted by finding the highest-scoring sentence. The fourth step then reduces the probability of each unit item appearing in future extracted sentences by multiplying each probability of each unit item in the last extracted sentence by itself. The implementation using terms as unit items first had a stop word list applied. The stop list was the same list used for the FreqDist summarizer. For the implementation using concepts, the UMLS Metathesaurus was used as the domain-specific resource. This was done to compare the SumBasic approach with our proposed FreqDist algorithm, which can also use concepts as unit items.

4.5.1.6 SWESUM

SweSum [37] is a multi-lingual summarizer for Swedish and English text. SweSum uses multiple features for scoring sentences, such as sentence position and numerical data identification. Sentences located earlier in a text are scored higher than sentences at the end of the text. Sentences containing numerical data are given additional weight. User-specified keywords can also be provided to boost sentence scores for those sentences containing the keywords. For the evaluation we used the online version located at http://swesum.nada.kth.se/index-eng-adv.html. The text type was set to 'Academic' and the summarization size was to 20%. No other parameters were set, and no domain specific knowledge sources were provided to the summarizer.

5. RESULTS

The results of the evaluation using ROUGE are shown in Tables 2 and 3. Each table is sorted in descending order based on the ROUGE score used. The best performing summarizer in each table is the first entry, while the lowest performing summarizer is listed as the last entry in each table. For the SumBasic and our FreqDist summarizer, two types of entries are listed: one entry using terms as unit items and the other entry using biomedical concepts as unit items.

5.1 ROUGE-2 Scores

Table 2 shows the ROUGE-2 scores for each summarizer. The best performing summarizes are the context-based SumBasic and our FreqDist. The FreqDist summarizer, when using Dice's coefficient for its similarity measure, outperforms all of the other summarizers using both terms and concepts as unit items. The performance of FreqDist using concepts and terms is close. This means that our FreqDist will also work well in a general domain that usually does not provide a way to find concepts due to lack of ontologies (or knowledge resources). The SumBasic summarizer performs better using terms rather than concepts, where the use of terms scored one percentage point better than the use of concepts. Our FreqDist summarizer performs best when using Dice's coefficient as the similarity measure between the summary and the source text. Dice is a measure of the common membership of unit items in the summary and source text. Other similarity measures, such as cosine, take into consideration not only membership, but also the weight (frequency) of each unit item. This leads us to conclude that our frequency distribution model approach (described in Section 3) requires no additional weighting of unit items to obtain good results. However, the use of frequency weights for comparing source text and candidate summaries also performs above both the baseline and general-purpose summarizers using Cosine and Unit Item Frequency. The use of frequency weights does not outperform the use of simple unit item membership.

Table 2. ROUGE-2 Scores for each summarizer

FreqDist-Term_Dice	0.22176
FreqDist-Concept_Dice	0.21997
SumBasic-Term	0.21112
FreqDist-Term_UnitFrequency	0.20707
SumBasic-Concept	0.20034
FreqDist-Concept_Cosine	0.19932
FreqDist-Concept_UnitFrequency	0.19932
MEAD	0.17629
FreqDist-Term_Cosine	0.17358
Baseline-Random	0.16396
AutoSummarize	0.15171
SweSum	0.15115
Baseline-Lead	0.13953
FreqDist-Concept_VectorSubtraction	0.11435
FreqDist-Concept_Euclidean	0.09236
FreqDist-Term_Euclidean	0.07516
FreqDist-Term_VectorSubtraction	0.05716

The worst performing summarizers are the ones based on the FreqDist algorithm using the Vector Subtraction and the Euclidean distance similarity measures (see Section 3 for details). These two similarity measures do not work well regardless of the unit items (i.e., terms or concepts). However, we note that in both methods, the use of concepts outperforms the use of terms

The MEAD summarizer, which employs a combination of features (see Section 4.5.1.3) to identify significant sentences, outperformed the Random sentence and Lead sentence baseline summarizers, and in fact fell just below the SumBasic and FreqDist summarizers in the performance table. The general purpose summarizers AutoSummarize and SweSum performed comparably, performing below the Random sentence baseline but above the Lead sentence baseline. This suggests to us that the simple use of frequency without either additional features (MEAD) or context sensitivity (SumBasic/FreqDist) is not effective with the summarization of biomedical text.

5.2 ROUGE-SU4 Scores

Table 3 shows the ROUGE-SU4 scores for each summarizer. In general, the ordering of the summarizer performance is about the same as in ROUGE-2. The best performing summarizers are the same as in ROUGE-2: our FreqDist and SumBasic. In both cases, the use of terms outperforms the use of concepts, but only by a margin of about 0.75 percentage points in both cases. Our FreqDist summarizer again performs best when using Dice's coefficient as the similarity measure between the summary and the source text. The Cosine and Unit Frequency also performed above the baseline and general-purpose summarizers. The use of the Vector Subtraction and Euclidean distance similarity methods with FreqDist was at the bottom of the performance list, as in ROUGE-2. The MEAD and FreqDist with Cosine similarity performed about the same using terms. The AutoSummarize and SweSum summarizers also performed closely, and were not much better than the Lead sentence summarizer. The Lead sentence baseline summarizer gave the worst performance when excluding the Vector Subtraction and Euclidean versions of FreqDist. The Random sentence baseline summarizer was in the middle of the performance table.

5.3 General Observations

It is interesting to note the baseline summarizer using random sentence selection performed nearly in the middle of the performance rankings for both ROUGE-2 and ROUGE-SU4. We are not sure how to interpret such high performance of random sentence selection. However, we do see that context sensitive methods such as SumBasic and our FreqDist methods significantly outperform the random baseline.

Excluding the FreqDist summarizers using the Vector Subtraction and Euclidean distance methods, the use of the lead sentences (i.e., Baseline-Lead in Tables 2 and 3) of a biomedical text generates the worst performance. This is important to note, because in text summarization work using the news genre, the lead sentence method often generates a very good summary [33]. This is because news stories are usually written so that the most important information appears at the beginning of the text, and the least important information at the end. However, in biomedical texts this assumption is invalid, as shown in Tables 2 and 3.

Using context-sensitive frequency methods, the use of concepts does not outperform the use of terms. However, terms and concepts perform closely. We find this valuable for building personalized summarizers that allow a user to select domain-specific concepts important to the user and then generate summaries for the user. It is easier for the user to select important concepts to summarize than important terms. This is because the concepts are defined for a domain, whereas terms are selected by author(s) of a paper and used in the text of the paper. To personalize a summary without domain-specific concepts, the user needs to know the important terms appearing in a text. In general, it is not easy for users to know terms in papers in advance before they read these papers.

Table 3. ROUGE-SU4 Scores for each summarizer

FreqDist-Term_Dice	0.12653
FreqDist-Concept_Dice	0.12070
SumBasic-Term	0.11673
FreqDist-Term_UnitFrequency	0.11664
SumBasic-Concept	0.10940
FreqDist-Concept_Cosine	0.10781
FreqDist-Concept_UnitFrequency	0.10781
FreqDist-Term_Cosine	0.09310
MEAD	0.09254
Baseline-Random	0.08001
AutoSummarize	0.07977
SweSum	0.07513
Baseline-Lead	0.07076
FreqDist-Concept_VectorSubtraction	0.05607
FreqDist-Concept_Euclidean	0.04356
FreqDist-Term_Euclidean	0.03429
FreqDist-Term_VectorSubtraction	0.02862

6. CONCLUSION

We proposed the frequency of domain-specific concepts as a feature for identifying salient sentences in biomedical texts. We presented an evaluation of several existing summarization systems to determine a performance baseline. We then evaluated a state-of-the-art frequency algorithm using both terms and concepts as item units to show the use of the frequency of concepts is as effective, and sometimes an improvement over, the use of frequency of terms. We developed a new algorithm based on frequency distribution modeling and evaluate it using terms as well as concepts. In either case, our frequency distribution algorithm outperforms a current state-of-the-art frequency-based algorithm at the cost of higher computational complexity. The use of concepts can be more useful in generating personalized summaries. An envisioned system allows a user to select domain-specific concepts important to the user, and then have the summarizer generate a summary where those concepts are more highly weighted than the concepts appearing in the source text.

There are several areas of future work. We would like to determine an optimum size of a biomedical text summary. While much work has been done in the news domain, little work has been done in the biomedical domain, where the source text size is

much larger and has multiple sections, each of which has varying importance to the overall content. We would also like to incorporate unit item frequency as an additional scoring feature into our existing summarization work based on lexical chaining of concepts [38]. For future evaluation work, we will include additional baseline summarizers to select sentences from throughout the text. For example, from the first sentence of each paragraph, each section, and so forth. Finally, we would like to use the FreqDist algorithm in the summarization of multiple biomedical source documents on the same topic.

7. REFERENCES

[1] S. D. Afantenos, V. Karkaletsis and P. Stamatopoulos, "Summarization from Medical Documents: A Survey," *Journal of Artificial Intelligence in Medicine,* vol. 33, pp. 157-177, 2005.

[2] A. D. Brooks and I. Sulimanoff, "Evidence-based oncology project," in *Surgical Oncology Clinics of North America* , vol. 11, Anonymous 2002, pp. 3-10.

[3] D. P. Jaques, *Surgical Oncology Clinics of North America: Prospective Randomized Clinical Trials in Oncology.* , vol. 11, W.B. Saunders Company, 2002, pp. 234.

[4] United States National Library of Medicine, "ClinicalTrials.gov," 2005.

[5] United States National Library of Medicine, "PubMed," 2005.

[6] A. M. Cohen and W. R. Hersh, "A survey of current work in biomedical text mining," *Brief Bioinform,* vol. 6, pp. 57-71, 2005.

[7] T. Berners-Lee, J. Hendler and O. Lassila, "The Semantic Web," *Sci. Am.,* vol. 284, pp. 34-43, May 2001.

[8] B. Chandrasekaran, J. R. Josephson and V. R. Benjamins, "What are ontologies and why do we need them?" *IEEE Intelligent Systems,* vol. 14, pp. 20-26, 1999.

[9] L. Reeve and H. Han, "Survey of semantic annotation platforms," in *Proceedings of the 20th Annual ACM Symposium on Applied Computing,* 2005.

[10] L. Reeve and H. Han, "A comparison of semantic annotation systems for text-based web documents," in *Web Semantics and Ontology* ,1st ed., vol. 1, D. Taniar and J. W. Rahayu, Eds. Hershey, PA USA: Idea Group, 2006.

[11] United States National Library of Medicine, "Unified Medical Language System (UMLS)," 2005.

[12] United States National Library of Medicine, "UMLS Metathesaurus Fact Sheet," 2004.

[13] United States National Library of Medicine, "MetaMap Transfer," 2005.

[14] H. P. Luhn, "The Automatic Creation of Literature Abstracts," *IBM Journal of Research and Development,* vol. 2, pp. 159-165, 1958.

[15] G. J. Rath, A. Resnick and R. Savage. The formation of abstracts by the selection of sentences. *American Documentation 2(12),* pp. 139-208, 1961.

[16] J. J. Pollock and A. Zamora, "Automatic Abstracting Research at Chemical Abstracts Service," *Journal of Chemical Information and Computer Sciences,* vol. 15, pp. 226-232, 1975.

[17] H. P. Edmundson, "New methods in automatic extracting," in I. Mani and M. T. Maybury, Eds. Cambridge, MA: MIT Press, 1999, pp. 23-42.

[18] E. D'Avanzo, B. Magnini and A. Vallin, "Keyphrase extraction for summarization purposes: The LAKE system at DUC-2004," in *Proceedings of the 2004 Document Understanding Conference,* 2004.

[19] E. Hovy and C. Lin, "Automated text summarization in SUMMARIST," in *Advances in Automatic Text Summarization* I. Mani and M. T. Maybury, Eds. Cambridge, MA: MIT Press, 1999, pp. 81-94.

[20] C. Fellbaum, *WORDNET: An Electronic Lexical Database.* The MIT Press, 1998,

[21] A. Nenkova and L. Vanderwende, "The impact of frequency on summarization," Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005-101, 2005.

[22] K. Sparck Jones, "Automatic summarizing: Factors and directions," in *Advances in Automatic Text Summarization* I. Mani and M. T. Maybury, Eds. Cambridge, MA: MIT Press, 1999, pp. 2-12.

[23] J. Carbonell and J. Goldstein, "The use of MMR, diversity-based reranking for reordering documents and producing summaries," in *SIGIR '98: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* 1998, pp. 335-336.

[24] G. Zipf, *Human Behavior and the Principle of Least Effort.* Cambridge, MA: Addison-Wesley, 1949,

[25] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval.* ,1st ed.Harlow, England: Addison-Wesley, 1999,

[26] L. R. Dice, "Measures of the amount of ecologic association between species," *Ecology,* vol. 26, pp. 297-302, 1945.

[27] S. Subhash, *Applied Multivariate Techniques.* ,1st ed.USA: John Wiley and Sons, 1996,

[28] D. L. Lee, H. Chuang and K. Seamons, "Document ranking and the vector-space model," *Software, IEEE,* vol. 14, pp. 67-75, 1997.

[29] C. Lin, "Recall-Oriented Understudy for Gisting Evaluation (ROUGE)," vol. 2005, April 13. 2005.

[30] C. Lin, "Looking for a few good metrics: Automatic summarization evaluation - how many samples are enough?" in *Proceedings of the NTCIR Workshop 4,* 2004.

[31] C. Lin and E. H. Hovy, "Automatic evaluation of summaries using N-gram co-occurrence statistics," in *Proceedings of 2003 Language Technology Conference (HLT-NAACL 2003),* 2003, pp. 71-78.

[32] National Institute of Standards and Technology (NIST), "Document Undertanding Conferences," vol. 2005, July 5, 2005.

[33] J. Goldstein, M. Kantrowitz, V. Mittal and J. Carbonell, "Summarizing text documents: Sentence selection and evaluation metrics," in *SIGIR '99: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* 1999, pp. 121-128.

[34] E. H. Hovy, "Automated text summarization," in *The Oxford Handbook of Computational Linguistics* R. Mitkov, Ed. Oxford: Oxford University Press, 2005, pp. 583-598.

[35] D. Radev, T. Allison, S. Blair-Goldensohn, J. Blitzer, A. Celebi, E. Drabek, W. Lam, D. Liu, J. Otterbacher, H. Qi, H. Saggion, S. Teufel, M. Topper, A. Winkel and Z. Zhang, "MEAD - a platform for multidocument multilingual text summarization," in *LREC 2004,* 2004.

[36] Microsoft Coporation, "Microsoft Word 2002," 2002.

[37] H. Dalianis, "SweSum - A text summarizer for swedish," NADA, KTH, Stockholm, Sweden, Tech. Rep. TRITA-NA-P0015, 2000.

[38] L. Reeve, H. Han and A. D. Brooks, "BioChain: Using lexical chaining methods for biomedical text summarization," in *Proceedings of the 21st Annual ACM Symposium on Applied Computing, Bioinformatics Track,* 2006.