Methods for Automatic Evaluation of Sentence Extract Summaries

G.Ravindra[+], N.Balakrishnan[+], K.R.Ramakrishnan[*]
Supercomputer Education & Research Center[+], Dept of Electrical Engg[*]
Indian Institute of Science, Bangalore 560012

***Abstract*— This paper describes three novel techniques to automatically evaluate sentence extract summaries. Two of these techniques called FuSE and DeFuSE evaluate the quality of the generated extract summary based on the degree of similarity to the model summary. They use a fuzzy set theoretic basis to generate a match score. DeFuSE is an enhancement to FuSE and uses WordNet based hypernymy structures to detect similarity between sentences at abstracted levels. The third technique focuses on quantifying the quality of an extract summary based on the difficulty in generating such a summary. Advantages of these techniques are described with examples.**

***Index Terms*— Collocation, Fuzzy set theory, s-norm operator, Summarization, WordNet**

I. Introduction

The Word Wide Web has revolutionized access, storage, search and retrieval of information. The vast amount of online data has introduced new paradigms into research in these areas. Although there is a huge amount of information available, the sheer volume has become an impediment to consumption/assimilation of such a vast body of information. To aid quick assimilation of information, machine based summarization of online news articles, books and journals will soon become part of a digital library hosting information. With improvements in automated summarization strategies, the need to evaluate and compare the efficiency of these strategies also becomes important. Methods to evaluate summaries can be classified into human evaluation methods and machine-based evaluation methods. In many cases human evaluation is aided by applications such as SEE 2.0 [1]. Evaluation is by comparing candidate (machine generated) summaries with reference/model (human generated) summaries and assigning scores. One of the problems with human evaluation is that of consistency. Two human judges may find it difficult to agree upon each other's judgement. Another problem is, a human judge may pass a different judgement if guidelines for evaluation are fuzzy. On the other hand automatic or machine based evaluation is always consistent with a judgement. The biggest handicap for automatic evaluators is that they lack the linguistic skills, moral and emotional biases which a human has. Although automatic evaluators are not perfect in evaluation, they are popular because consistent and quick evaluation of a large number of summaries is possible.

II. Review of summary evaluation techniques

Scoring during evaluation, is typically based on the extent to which content in the machine generated summary matches with the model summary. A match could be classified as *fully* matches, *almost* matches, *partially* matches or *hardly* matches with appropriate weights assigned to each type of match. Alternatives are to use recall score [2] and coverage score [3]. Automatic evaluators which use specific matching and scoring techniques include BLEU with brevity penalty [4],[5], BLEU with brevity bonus [6], "Longest Common Subsequence (LCS)" match, [7], "Normalized Pairwise LCS" [8], ROUGE n-gram and "Weighted LCS" match [7]. Although these techniques concentrate on matching and scoring segments of text between machine and model summaries, they do not handle specific issues such as multiple matches, subsumption and hypernymous/synonymous word usage.

In this paper we propose novel automatic summary evaluation techniques called FuSE and DeFuSE. FuSE (Fuzzy Summary Evaluator) uses fuzzy set theory based scores for sentences using matches between word collocations in machine and model summaries. DeFuSE on the other hand enhances the reliability of FuSE by exploiting WordNet [9], [10], [11] hypernymy structure of words. Further, we also propose a complexity score which quantifies how difficult it is to generate a summary of a particular accuracy. The remaining sections of this paper are organized as follows: in section-3 FuSE is described along with the mathematical basis for the scheme, section-4 describes DeFuSE, section-5 introduces the complexity score and the paper is concluded in section-6. Throughout this paper we use "summary" and "extract" interchangeably.

III. Fuzzy Summary Evaluator: FuSE

FuSE evaluates summaries by representing the model extract summary as a fuzzy set. Each sentence in the candidate summary has a membership grade in this set (please refer to ([12]) for a detailed discussion on Fuzzy set theory). Further it is assumed that every sentence in the candidate extract has a membership grade associated with every sentence in the model extract. Hence, the membership grade of a sentence in the model extract is the union of its sentence level membership grades.

A. Membership grades

Let $R = \{r_1, r_2, .., r_{|R|}\}$ be a model extract consisting of $|R|$ sentences and $C = \{c_1, c_2, ..., c_{|C|}\}$ be a candidate extract consisting of $|C|$ sentences. A sentence $c_i \in C$ is said to have some similarity with every sentence in R,

A PART-TIME MAINTENANCE WORKER ACCUSED OF KILLING FOUR PEDESTRIANS IN GLENDALE HAS BEEN ORDERED TO STAND TRIAL LATER THIS MONTH IN PASADENA SUPERIOR COURT.
A PART-TIME MAINTENANCE WORKER HAS BEEN ACCUSED OF KILLING FOUR PEDESTRIANS IN GLENDALE. HE HAS BEEN ORDERED TO STAND TRIAL LATER THIS MONTH IN PASADENA SUPERIOR COURT.

Fig. 1. A case of subsumption

similarity being a number in the range $[0, 1]$. This similarity is called the membership grade. As we intend to evaluate an extract consisting of sentences, the membership grade becomes a measure for sentence similarity. FuSE uses collocations (extracted using a window length of two words) for comparing sentence similarity. As membership grade can be looked at as a measure for similarity, we consider the use of Hamming distance as the candidate similarity measure. Although there are a number of ways to assign membership grades ranging from the use of intuition to genetic algorithms, Hamming distance is chosen as it is simple and ideal for a linguistic application where we need to count number of matching units.

Hamming distance membership grade counts the number of matching collocations in a sentence and then normalizes the count by the length of the sentence. For example, the first sentence of the right-hand side summary in fig-1 has 7 collocations which are also found in the left-hand side summary. The sentence length is also 7 collocations long. Hence, all collocations are found and the sentence should get a score of 1. Similarly the second sentence also should get a score of 1. Such a scoring scheme is possible if the membership grade $\mu_{r_j}(c_i)$ is defined as

$$\mu_{r_j}(c_i) = \frac{|c_i \cap r_j|}{|c_i|} \tag{1}$$

Here, $\mu_{r_j}(c_i)$ is the membership grade of the sentence c_i in the sentence r_j.

1) Fuzzy precision score: Let every sentence $r_j \in R$ be considered as a fuzzy set. As a result, R now becomes a collection of fuzzy sets and sentence $c_i \in C$ has a membership grade in each of these fuzzy sets. Let $\mu_{r_j}(c_i)$ be the membership grade of the sentence c_i in the fuzzy set r_j. The reference summary can be written as $R = \cup_{j=1}^{j=|R|} r_j$, a union of fuzzy sets. In classical set theory, the membership grade of an element in a set is 0 or 1. Hence, the membership grade of an element in the union can be written as

$$\mu_R(c_i) = \begin{cases} \max\limits_{j=1..|R|} \left(\mu_{r_j}(c_i)\right) = 0; & if\ \mu_{r_j}(c_i) = 0\ \forall\ c_i \\ \max\limits_{j=1..|R|} \left(\mu_{r_j}(c_i)\right) = 1; & if\ \mu_{r_j}(c_i) = 1\ for\ some\ c_i \end{cases} \tag{2}$$

The fuzzy set union can also be written as mentioned in (2), but the membership grade in the union need not be only 0 or 1. Further an S-Norm operator (union operator) $\bigvee$ can be defined to replace the max operator used in (2). Then fuzzy precision can be defined as

$$p_F = \frac{\sum_{i=1}^{i=|C|} \left(\bigvee_{j=1..|R|} \mu_{r_j}(c_i) \right)}{\sum_{i=1}^{i=|C|} \left(\bigvee_{j=1..|C|} \mu_{c_j}(c_i) \right)} \tag{3}$$

2) Fuzzy recall score: Let every sentence $c_i \in C$ be considered as a fuzzy set. As in the case of precision computation, C now becomes a collection of fuzzy sets and sentence $r_j \in R$ has a membership grade in each of these fuzzy sets. Let $\mu_{c_i}(r_j)$ be the membership grade of the sentence r_j in the fuzzy set c_i. The candidate summary can be written as $C = \cup_{i=1}^{i=|C|} c_i$, a union of fuzzy sets and, the fuzzy recall score can be computed as

$$r_F = \frac{\sum_{j=1}^{j=|R|} \left(\bigvee_{i=1..|C|} \mu_{c_i}(r_j) \right)}{\sum_{j=1}^{j=|R|} \left(\bigvee_{i=1..|R|} \mu_{r_i}(r_j) \right)} \tag{4}$$

3) Computing F-score: Using (3) and (4) the fuzzy f-score is computed as

$$f_{score} = \frac{2 \times p_F \times r_R}{p_F + r_F} \tag{5}$$

B. The union operator

The union operator should satisfy all the requirements of fuzzy set theory and at the same time exhibit properties which result in a correct automatic summary evaluation. We choose to use Frank's union operator instead of the Max union operator as the s-norm operator. The reason for this choice shall become clear soon, but we shall first see how the Max union operator performs.

1) MAX s-norm operator : One of the most popular union operators is the max operator. The use of max operator is valid both in the classical set theory union and fuzzy unions. In a matrix Φ, of membership grades, an element Φ_{ij} is the membership grade of the i^{th} sentence of the reference summary, in the j^{th} sentence of the candidate summary. This can also be written as $\Phi_{ij} = \mu_{c_j}(r_i)$. Hence elements along the i^{th} row correspond to the membership grades of this sentence in $|C|$ fuzzy sets, where each of these fuzzy sets represents a sentence in the candidate summary. Similarly the elements along the j^{th} column correspond to the membership grades of the corresponding sentence of the candidate summary, in $|R|$ fuzzy sets, where each of these fuzzy sets represents a sentence in the reference summary. The membership grade of the i^{th} sentence of the candidate summary can be computed using the max-union operator as $\mu_R(c_i) = \max_{j=1..|R|}\left(\mu_{c_j}(t_i)\right)$. Similarly to find the membership grade of the i^{th} sentence of the reference summary in the fuzzy union, the relation $\mu_C(r_i) = \max_{j=1..|C|}\left(\mu_{c_j}(r_i)\right)$ can be used. Behaviour of

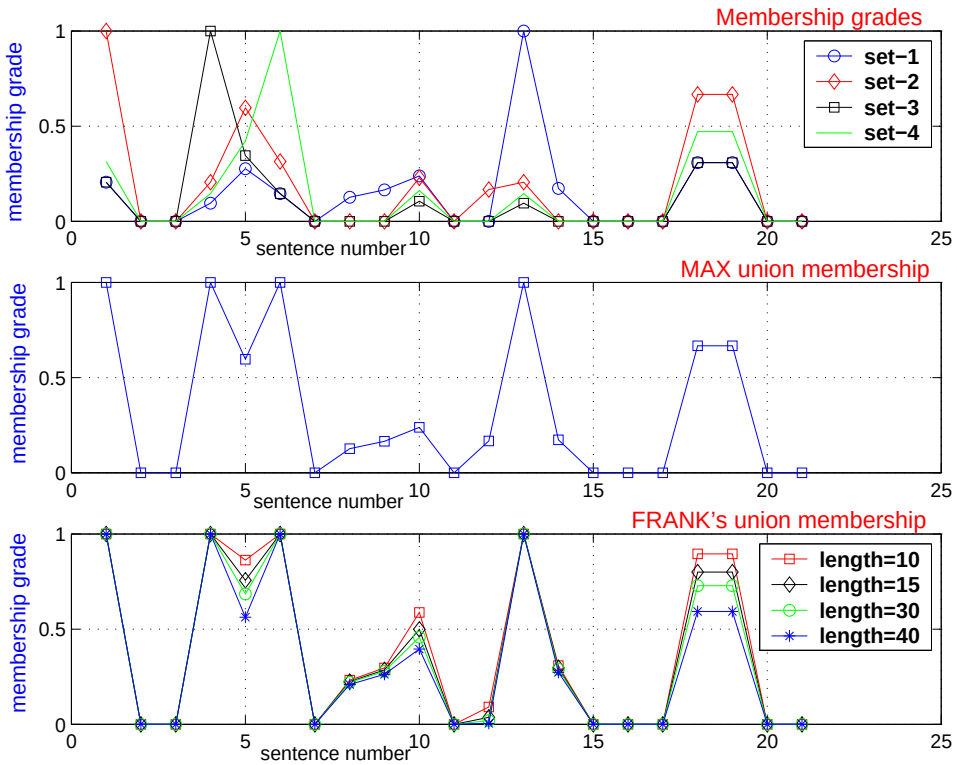

Fig. 2. Behavior union operators with changing sentence length

the *max* s-norm operator is shown in fig.2. The top most plot in fig.2 shows the membership grades of 21 sentences in 4 fuzzy sets (4 sentences of the reference summary). It can be observed that membership grade for some sentences is 1, in at least one of the sets. This is likely to be the case when a sentence matches exactly with another sentence, but also has partial matches with other sentences. There are also sentences whose membership grades are always less then 1 and have matches in all the 4 sets. This is likely to be the case when a single sentence has parts matching with multiple sentences. The second graph in the figure shows the resultant membership grade of these 21 elements after using the *max* s-norm operator. If we use the *max* s-norm operator for evaluating a summary, the score for a sentence in the model summary can never exceed its maximum membership grade. This becomes a problem when a candidate sentence is a union of collocations present in more than two sentences of the model summary. Hence a union operator which combines membership grades to produce a resulting membership grade greater than the individual maximum is needed.

2) FRANK's s-norm operator: The max union operator does not take into consideration the distribution of membership grades in various sets. The implication of this observation becomes evident from the following example. Consider the reference summary containing sentence R1 and the candidate summary containing two sentences C1 and C2:

R1: HURRICANE GILBERT STRUCK THE SOUTHERN COAST OF CUBA YESTERDAY CAUSING SEVERE DAMAGE TO THE COASTAL BELT.

C1: HURRICANE GILBERT STRUCK SOUTHERN CUBA.

C2: SEVERE DAMAGE WAS REPORTED ALONG THE COASTAL BELT.

Using Hamming distance membership grade assignment with a collocation window length of 2 words, we find that membership grade of C1 in R1 is $\frac{2}{4}$ and C2 in R1 is $\frac{2}{5}$. Using the *max* union operator, the membership grade of C1 and C2 together in R1 (precision score) is max($\frac{2}{4}$, $\frac{2}{5}$) which is 0.50. Similarly the membership grade of R1 in C1 and C2 is $\frac{2}{11}$, which means the membership grade of the reference in the candidate (recall score) is 0.18. If f-score is computed using these values the candidate summary gets a score of 0.26. Now if C1 and C2 were to be combined into a single sentence, then the precision and recall using *max* union operator would be $\frac{4}{9}$ and $\frac{4}{11}$ respectively. The f-score would have been 0.396 which is greater then the previous case. This means we require a union operator which can combine membership grades such that the combined membership grade is greater then the maximum of individual membership grades. Further, if the membership grade is a function of sentence length it would be an added advantage.

There are a number of union operators which combine individual membership grades to produce a result greater than any one of them. Most interesting of them, especially from the summary evaluation point of view is the Frank's Union operator [13]. Given an element x with membership grades $\mu_A(x)$ and $\mu_B(x)$ in two fuzzy sets A and B, the Frank's Union operator is defined by the relation

$$\left(\mu_A(x) \bigvee \mu_B(x)\right) = 1 - \log_F \left[1 + \frac{\left(F^{1-\mu_A(x)} - 1\right)\left(F^{1-\mu_B(x)} - 1\right)}{F-1}\right] \quad (6)$$

This union operator non-linearly combines individual membership grades and at the same time confirms to all the laws of fuzzy union. The base of the logarithm, F, plays an important role in the combination process and is defined as $F \neq 1, F > 0$. To adopt Frank's union operation to summary evaluation, the value of F was defined as

$$F = \exp\left(-\tau \times m \times \frac{S_L}{\max_L}\right) \quad (7)$$

where τ is a damping factor, m is the mean of the non-zero membership grades of a sentence x, S_L is the length of x in terms of the basic units (collocations in this case) and $\max_L$ is the length (number of collocations) of the longest sentence in the set of sentences being evaluated. We choose $\tau = 10$ based on the observation that $\max_L$ usually is not more than 20 collocations long. F can become equal to 1 only when m is zero and this can happen if a sentence has a zero membership grade everywhere. Such sentences are programatically eliminated from the evaluation process.

Fig.3 shows how the membership grade of a sentence in the fuzzy union varies for different values of τ. For $F > 1$, a fixed m and fixed $\max_L$, we find that when sentence length S_L is high, the membership grade in the union is also high (first plot in fig.3). But we would like to have a lower membership grade, if the sentence length is more and the mean non-zero membership grade is fixed. Hence the solution is to choose $F < 1$ by using the exponential shown in (7). The behaviour of (7) for different values of τ is shown in the second plot in Fig-3.

The result of using the Frank's s-norm operator can be observed in fig.2. The third graph in fig.2 shows the result of union operation using the $Frank's$ union operator with exponential base given by (7) for various sentence lengths. For example, sentence number 5 has membership grades $\{0.6, 0.4, 0.34, 0.28\}$ in the 4 fuzzy sets. The max union results in a membership grade of 0.6 for this sentence in the reference summary set and it is independent of the length of this sentence. But the $Frank's$ union operation, results in membership values of $\{0.86, 0.75, 0.68, 0.56\}$ for sentence lengths of $\{10, 15, 30, 40\}$ collocations respectively. This means sentences which are shorter, having the same membership grade and having a higher mean value computed using non-zero memberships, get a higher score after the union operation. This is very useful when the sentence being evaluated is a super set of a number of sentences in the reference set. As already mentioned, the fuzzy summary evaluator built using the Frank's union operator is referred to as FuSE (Fuzzy Summary Evaluator).

C. Comparison of FuSE with ROUGE-v1.2.1

FuSE was compared with ROUGE-v1.2.1 by taking different test cases. There is no effective way to evaluate an automatic evaluator except by comparing with a competitive method or by human judgment. ROUGE has been extensively evaluated by comparing with human judgment and hence we use this evaluator to see how similar FuSE is in its scoring mechanism.

ROUGE has 3 classes of recall based scores, namely ROUGE-n, ROUGE-LCS and ROUGE-WLCS. ROUGE-n is an n-gram score where any two sentences having the same set of n-gram sequences are assumed to have contributed to a match. ROUGE-LCS is the LCS-based score while ROUGE-WLCS is a non-linearly weighted LCS score.

As the first case for comparison, a machine generated summary which is an exact replica of a human summary was used and recall-based ROUGE evaluation system was considered. As every sentence in the machine summary will have an exact match with a sentence in the model (human generated) summary, we expect ROUGE scores to be 1 and FuSE precision, recall and F-scores to be 1 as well. But it was found that ROUGE-WLCS does not have a normalized non-linear weighting, as a result of which it gave a recall score of 0.35 for this test case. The exact values of these scores are listed in table-I.

Evaluation Type	Score Value
ROUGE-1	1
ROUGE-2	1
ROUGE-3	1
ROUGE-4	1
ROUGE-LCS	1
ROUGE-WLCS	**0.35764**
FuSE Precision	1
FuSE Recall	1
FuSE f-score	1

TABLE I
EVALUATION SCORES WHEN REFERENCE AND CANDIDATE SUMMARIES ARE IDENTICAL

As a second case for comparison, sentences which are almost similar in word composition but not exactly the same were considered (selected from the DUC 2002 data set). For example consider the following model summary (with stemmed words)

MODEL: JAMMU KASHMIR IS THE ONLY STATE WITH A MOSLEM MAJORITY IN PREDOMINANT HINDU INDIA. INDIA ACCUSE PAKISTAN OF ARM MILITANT.

and the following candidate machine generated summaries

CAND-1: INDIA DOMINATE BY HINDU ACCUSE PAKISTAN OF TRAIN AND ARM KASHMIR MILITANT FIGHT FOR SECESSION OF JAMMU KASHMIR A MOSLEM MAJORITY STATE

CAND-2: HINDU DOMINATE INDIA ACCUSE PAKISTAN OF ARM AND TRAIN KASHMIR MILITANT FIGHT FOR SECESSION OF JAMMU KASHMIR INDIA ONLY STATE WITH MOSLEM MAJORITY

CAND-3: ARM AND TRAIN KASHMIR MILITANT FIGHT FOR SECESSION ACCUSE MAJORITY OF MOSLEM IN JAMMU KASHMIR STATE IN INDIA AND HINDU IN PAKISTAN FOR THE PRESENT CONDITION

In this example we observe that the candidate summaries have only one sentence and is a union of information present

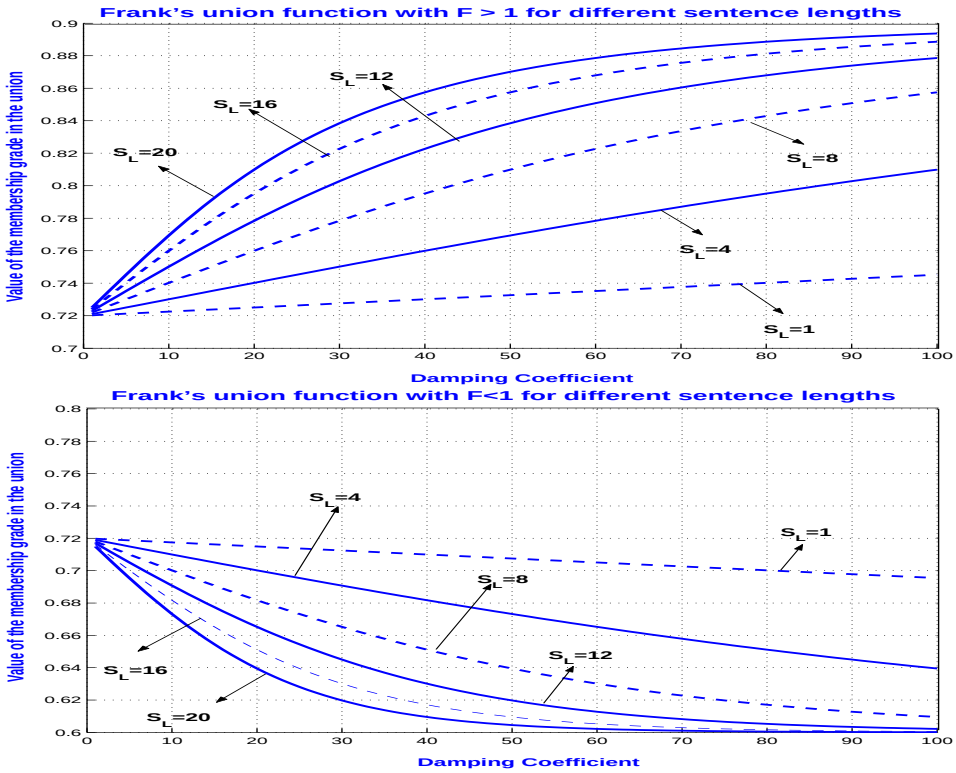

Fig. 3. Membership grade using Frank's s-norm operator

Scoring Type (F-score)	Cand-1	Cand-2	Cand-3
ROUGE-1	0.65	0.69	0.63
ROUGE-2	0.18	0.3	0.18
ROUGE-3	0	0.12	0.09
ROUGE-4	0	0	0
ROUGE-LCS	0.51	0.56	0.4
ROUGE-WLCS	0.25	0.29	0.21
FuSE (window=2)	0.23	0.36	0.07
FuSE (window=3)	0.22	0.324	0.09

TABLE II
COMPARISON OF EVALUATORS FOR CASE-2

in the two sentences of the model summary for the first two candidates, and is completely different information with respect to the third candidate. Further, the candidate summaries are same in word composition but word order has changed. Table-II shows scores produced by ROUGE and FuSE of this example. One of the drawbacks in ROUGE can be observed in the results for candidates 2 and 3 where it can be observed that the longest common sub-sequence defaults to a sequence of words and hence becomes a bag of words. Word order is not of importance and candidates 2 and 3 are assigned scores which are relatively closer. On the other hand FuSE f-score uses collocations extracted with a window length of two and hence can easily capture not only longer sub-sequences but can also detect completely different meaning (owing to drastic change in word order). The two cases described so far show that ROUGE-WLCS penalizes extracts more than required and ROUGE-LCS score can become the same as ROUGE-1 due to the bag of words observation just described.

As the third case for comparison, relationship between ROUGE and Fuzzy f-score values was investigated. As ROUGE scores are recall centric, a precision score was obtained by interchanging the candidate and reference summaries while passing as arguments to the ROUGE program. The resulting score can be considered as the precision score and a f-score was computed. Fig.4 shows f-scores produced by ROUGE-LCS, ROUGE-WLCS, FuSE and Exact sentence match. The two graphs in this figure show that ROUGE-WLCS and ROUGE-LCS scores vary similarly, suggesting that the difference primarily exists in the scaling factor. On the other hand ROUGE and FuSE can assign a non-zero score to summaries unlike exact sentence match. This means if a candidate summary does not have any sentences matching exactly with the model summary, it can still receive a non-zero score owing to partial matches. Further, FuSE seems to be different from ROUGE especially in cases where the match percentage is relatively low and seems to vary like ROUGE where the percentage of match is high.

Fig.5 shows a comparison of FuSE (collocation window size of 2) with ROUGE-2. As a 2-gram match is a tighter matching criterion than a 1-gram match and avoids LCS match defaulting to a 1-gram match, it is more reliable. At the same time its is not as strict as a 4-gram match and this is helpful when two sentences are almost similar in word composition but not exactly the same. It can be observed in fig.5 that in many cases ROUGE-2 and FuSE produce similar evaluation although having different scales. But there are cases where FuSE has assigned a relatively high score to some summaries and in some cases a lower score. This is because of the ability

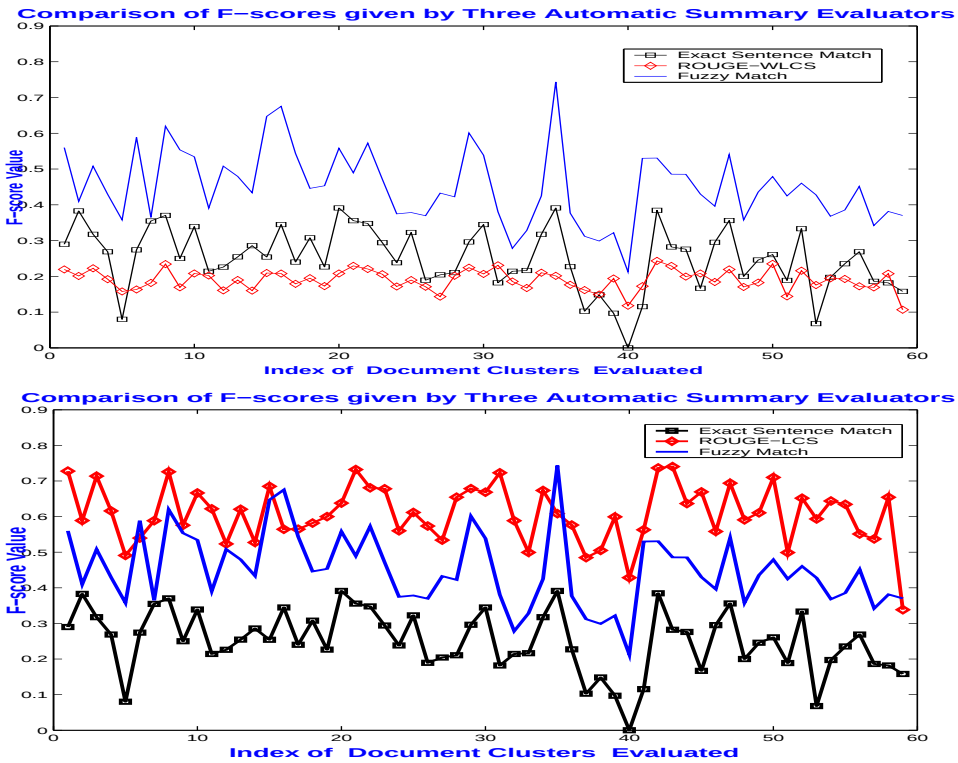

Fig. 4. Comparison of f-scores of ROUGE, Fuzzy and Exact-match evaluators

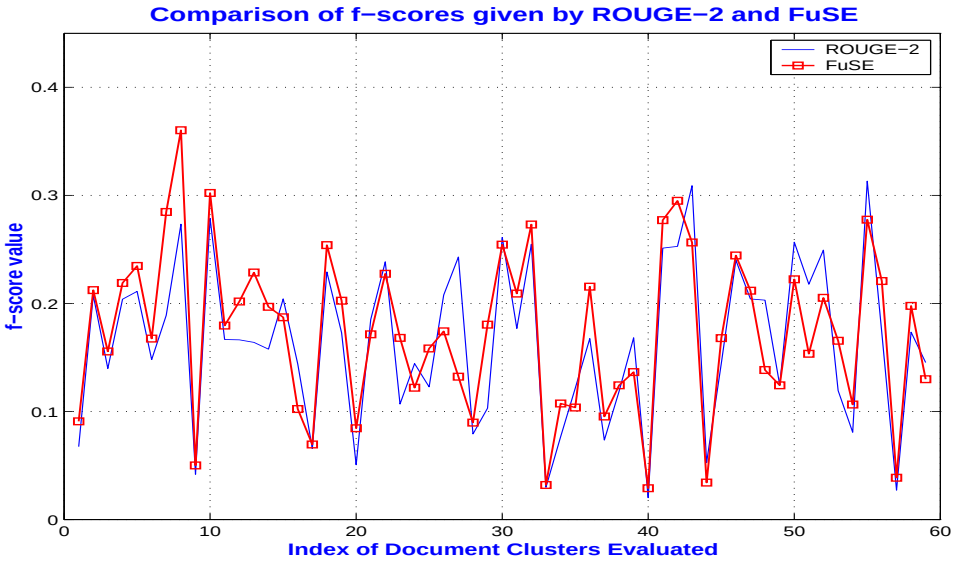

Fig. 5. Comparison of f-scores of ROUGE-2 and FuSE

of FuSE to evaluate sentence similarity and hence summary similarity, taking into account the distribution of collocations across sentences along with sentence length.

IV. Dictionary-enhanced Fuzzy Summary Evaluator: DeFuSE

One of the biggest problems encountered in automatic summary evaluation is to account for synonyms and conceptual information. What constitutes a concept is very difficult to identify. DeFuSE is an extension to FuSE and it uses a WordNet based dictionary to identify synonyms and hypernymy structure of words before assigning a membership grade to sentences. Words are classified as nouns, verbs, adjectives, adverbs and stemming of words is automatically performed. This is because WordNet returns results after internally converting words to their base forms. If a word can have both adjective sense and noun sense, its adjective sense is taken and if a word has both noun and verb sense, its noun sense is taken

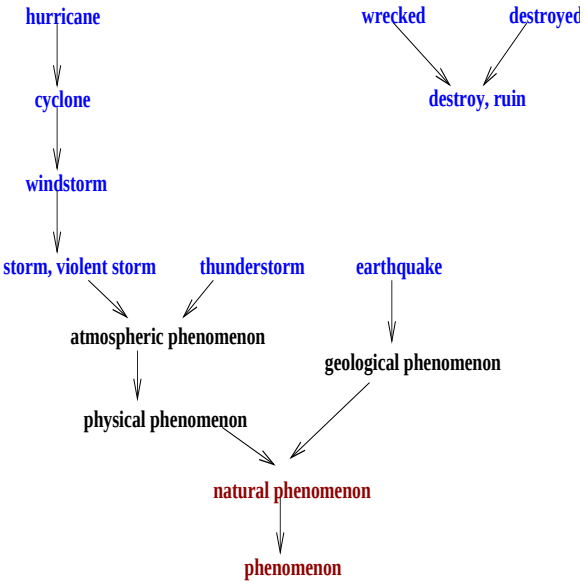

Fig. 6. Example hypernymy for the concept of '*natural phenomena*'

into consideration. Nouns and verbs exhibit hypernymy and this helps to determine similarity of any two sentences with a higher degree of accuracy. For example consider sentences

C1: HURRICANE GILBERT DEVASTATED DOMINICAN REPUBLIC AND PARTS OF CUBA

C2: TROPICAL STORM GILBERT DESTROYED PARTS OF HAVANA

The two sentences convey similar meaning but any distance measure which uses only word occurrence would definitely not indicate an exact match. The problem is due to synonymy and hypernymy. We can always argue that these two sentences almost mean the same at some level of abstraction by saying that a hurricane is a tropical storm, and destruction of parts of Havana also means destruction of parts of cuba. The aim of DeFuSE is to capture these abstractions while assigning fuzzy scores. The figure-6 shows an example of a hypernymy tree based abstraction for the words 'hurricane' and 'thunderstorms'. It can be observed that 'hurricane' is a type of 'cyclone' which in turn is an 'atmospheric phenomenon'. Similarly a 'thunderstorm' is also an 'atmospheric phenomenon'. Hence 'hurricanes' and 'thunderstorms' are synonyms at some level of abstraction which in this case is 'atmospheric phenomenon'. There is no difference between FuSE and DeFuSE except that the latter uses WordNet expansions for words before it can use the same fuzzy union (eq-6). Every sentence is translated to its WordNet equivalent. This is done by the following simple rules

- nouns which are also adjectives are treated as adjectives
- verbs which are also nouns are treated as nouns
- in a sentence if a word happens to be an adjective or an adverb, the word is replaced with its synonym
- if a word happens to be a noun, then its hypernymy based abstraction three levels below the highest abstraction is chosen.
- if a word happens to be a verb, then its hypernymy based abstraction one level below the highest abstraction is chosen.
- always the first sense given by WordNet is used

Nouns can have very deep trees and verbs usually have shallow depths. Hence we choose 3 levels above the maximum depth for nouns and 1 level above the maximum depth for verbs. A single word can have multiple WordNet senses, and hence we choose only the first WordNet sense. We are not interested in the accuracy of the sense but only interested in the choice of sense being consistent. Another advantage which we have while using WordNet is the ability to detect word combination. Take for example the words 'Dominican' and 'republic'. When they occur separately their meaning is different from that when considered together as in "Dominican republic". Detecting this type of word combination is possible using WordNet and DeFuSE programatically tries to group words so that they produce a more complete meaning. The concept of finding abstracted versions of sentences for evaluation can be better understood by taking note of the following WordNet abstractions:

- dominican republic-> (country, state, land) => (administrative district, administrative division, territorial division) => (district, territory) => **region** => location => entity
- cuba-> (country, state, land) => (administrative district, administrative division, territorial division) => (district, territory) => **region**=> location=> entity
- hurricane-> cyclone=> windstorm=> storm, violent storm=> atmospheric phenomenon=> **physical phenomenon**=> natural phenomenon=> phenomenon
- storm-> atmospheric phenomenon=> **physical phenomenon**=> natural phenomenon=> phenomenon
- devastated-> **(destroy, ruin)**
- havana-> national capital=> capital=> seat=> center, centre, middle, heart, eye=> area, country=> **region**=> location=> entity

In the above mentioned abstractions, the first word (before the '->' symbol) is the actual word occurring in the sentence. Each set of words enclosed between the '=>' symbol is the corresponding abstraction at that level. Abstractions shown in bold font are those selected by DeFuSE. The first three words are nouns and we find that the words 'Dominican republic' and 'cuba' mean the same, i.e they are 'regions'. The fourth word, i.e 'storm', can be a noun as well as a verb. As we choose its noun form this word means 'physical phenomenon' and this is the same as 'hurricane's' abstraction. Hence words 'storm' and 'hurricane' are treated as the same. The fourth word, i.e 'devastated', is a verb and it has only one level of abstraction and hence this level is chosen. The fifth word, i.e 'havana' also translates to the word 'region' and hence 'cuba' and 'havana' both are translated to mean 'region'. Hence the sentence

C1: HURRICANE GILBERT DEVASTATED DOMINICAN REPUBLIC AND PARTS OF CUBA

becomes

C1': (PHYSICAL PHENOMENON) GILBERT (DESTROY,RUIN) (REGION) AND PARTS OF (REGION)

and the sentence

C2: TROPICAL STORM GILBERT DESTROYED PARTS OF HAVANA

becomes

C2': TROPICAL (PHYSICAL PHENOMENON) GILBERT DESTROYED PARTS OF (REGION)

In terms of **C1'** and **C2'**, sentences **C1** and **C2** are closer in meaning than otherwise. **C1'** and **C2'** mean that "a physical phenomenon called Gilbert had some impact on parts of some region". Hence at an abstracted level, sentences C1 and C2 are similar.

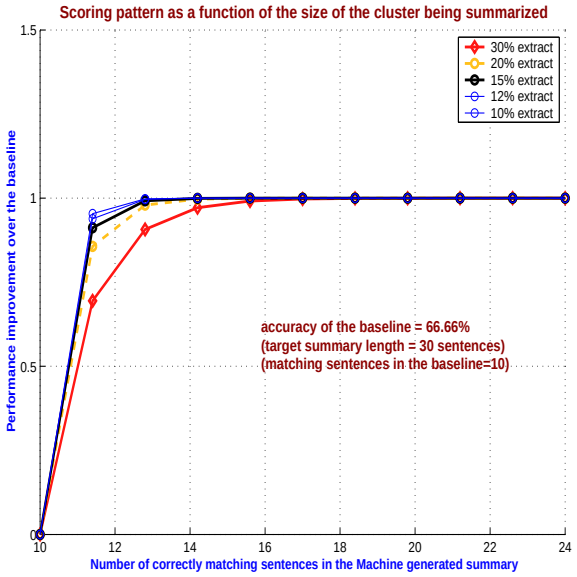

Fig. 7. Scoring pattern as a function of cluster size

V. COMPLEXITY SCORE

One of the biggest problems in summarization research is to find a method to compare the performance of summarizers such that the complexity of the task is also taken into account. Although there have been a few automatic evaluation metrics, it is still difficult to say whether a new summarization algorithm is better than previous methods. For example if a summarizer is 90% accurate and a new algorithm is 91% accurate, it is very difficult to say whether the new method is significant in performance. Depending on how performance improvement is measured, we might get different impressions about any algorithm. For example, if we measure improvement in terms of percentage increase in accuracy, we get an improvement of $\frac{91-90}{90} \times 100 = 1.11\%$. But if we measure improvement in terms of percentage decrease in error, we get an improvement of $\frac{(100-90)-(100-91)}{(100-90)} \times 100 = 10\%$. Hence quoting performance improvement seems to be a misnomer. To avoid such confusion we propose the use of a complexity score, which takes into account the difficulty in performing the job of summarization. Such a score is very difficult to derive for any arbitrary application, but the nature of extractive summarization easily allows us to do so.

An extractive summarizer can be viewed as a system which is presented with a number of choices, out of which it chooses the best possible according to its capacity. The summarizer is given n sentences out of which it is expected to choose h sentences, such that these h sentences form the best possible summary. Now if a summarizer chooses m_1 sentences out of which l_1 sentences are accurate (as expressed by a human judge) and $m_1 - l_1$ sentences are inaccurate, then probability of such a choice can be written using the binomial

$$ {}^{n}c_{m_1} \left(\frac{h}{n}\right)^{l_1} \left(1-\frac{h}{n}\right)^{m_1-l_1} $$

Suppose we are given a reference summary generated by human judges containing h sentences, then the difficulty in generating such a reference summary can be quantified by ${}^{n}c_{h} \left(\frac{h}{n}\right)^{h}$. The closer a machine generated summary is to the human summary, better will be its performance. Hence the term ${}^{n}c_{m_1} \left(\frac{h}{n}\right)^{l_1} \left(1-\frac{h}{n}\right)^{m_1-l_1} - {}^{n}c_{h} \left(\frac{h}{n}\right)^{h}$ can be used as a measure for the deviation of the machine summary from the human summary and it accounts for the difficulty in performing the job of selecting correct set of sentences. Now let there be another summarizer (baseline) whose deviation from the reference is given by ${}^{n}c_{m_2} \left(\frac{h}{n}\right)^{l_2} \left(1-\frac{h}{n}\right)^{m_2-l_2} - {}^{n}c_{h} \left(\frac{h}{n}\right)^{h}$. If the baseline is farther away from the reference than the target summarizer, then the relative improvement of the target over the baseline can be computed as

$$ \frac{{}^{n}c_{m_2} \left(\frac{h}{n}\right)^{l_2} \left(1-\frac{h}{n}\right)^{m_2-l_2} - {}^{n}c_{m_1} \left(\frac{h}{n}\right)^{l_1} \left(1-\frac{h}{n}\right)^{m_1-l_1}}{{}^{n}c_{m_2} \left(\frac{h}{n}\right)^{l_2} \left(1-\frac{h}{n}\right)^{m_2-l_2} - {}^{n}c_{h} \left(\frac{h}{n}\right)^{h}} \times 100 $$

If $m_1 = m_2 = h$, then the relative improvement can be written in the form of a complexity score given by

$$ \frac{\left(\frac{h}{n}\right)^{l_2} \left(1-\frac{h}{n}\right)^{h-l_2} - \left(\frac{h}{n}\right)^{l_1} \left(1-\frac{h}{n}\right)^{h-l_1}}{\left(\frac{h}{n}\right)^{l_2} \left(1-\frac{h}{n}\right)^{h-l_2} - \left(\frac{h}{n}\right)^{h}} \times 100 \qquad (8) $$

Does this evaluation really reflect the complexity at hand? It does so, and is depicted in figure-7.

The figure depicts a case where the performance of an automatically generated summary is compared to the performance of a baseline summary for different document cluster sizes measured in terms of the total number of sentences. The x-axis refers to the number of sentences in the machine-based summary having a match in the model human summary. The human summary is 30 sentences in size and the baseline has a 66.66% match. The machine-based summary has a match between 66.66% and 80%. It can be observed that when the compression is low (30% extract), the performance of the automatic summarizer is poorer (with respect to the baseline) as compared to the case where the compression is higher (20% extract). This clearly shows that the complexity metric given by (8) takes into account the complexity involved in generating a summary in terms of number of sentences to select from. For small number of sentences, it is lot more easy to generate a good summary as the extract generator has limited number of choices.

VI. Conclusion

This paper presented three novel methods which can be used to automatically evaluate sentence level extract summaries. A fuzzy set theoretic approach called FuSE and its enhanced version called DeFuSE were described. Performance of FuSE was compared to ROUGE and the WordNet based enhancements which DeFuSE uses was described with examples. Finally a complexity based evaluation scheme was presented. This scheme accounted for the difficulty in generating a sentence extract summary of a particular size and accuracy.

References

[1] C.-Y. Lin, "Summary Evaluation Environment," 2001, http://www.isi.edu/~cyl/SEE.

[2] I. Mani, D. House, G. Klein, L. Hirshman, L. Orbst, T. Firmin, M. Chrzanowski, and B. Sundheim, "The TIPSTER SUMMAC Text Summarization Evaluation," The Mitre Corporation, McLean, Virginia, Tech. Rep. MTR 98W0000138, 1998.

[3] R. L. Donaway, K. W. Drummey, and L. A. Mather, "A Comparison of Rankings Produced by Summarization Evaluation Measures," in *Proceedings of the Workshop on Automatic Summarization at the 6th Applied Natural Language Processing Conference and the 1st Conference of the North American Chapter of the Association for Computational Linguistics*, U. Hahn, C.-Y. Lin, I. Mani, and D. R. Radev, Eds. Association for Computational Linguistics, April 30 2000, pp. 69–78.

[4] K. Papineni, S. Roukos, T. Ward, and W. Zhu, "BLEU: A Method for Automatic Evaluation of Machine Translation," in *Proceedings of the 4Oth Meeting of the Association for Computational Linguistics*, Philadelphia, USA, 2002, pp. 311–318.

[5] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A Method for Automatic Evaluation of Machine Translation," IBM, Research Report RC22176, 2001.

[6] C.-Y. Lin and E. Hovy, "Automatic Evaluation of Summaries Using N-gram Co-Occurrence Statistics," in *Proceedings of the Human Language Technology Conference*, 2003.

[7] H. Saggion, D. Radev, S. Teufel, and W. Lam, "Meta-Evaluation of Summaries in a Cross-Lingual Environment Using Content-Based Metrics," in *Proceedings of COLING-2002*, Taipei, Taiwan, 2002.

[8] D. R. Radev, S. Teufel, H. Saggion, W. Lam, J. Blitzer, A. Çelebi, H. Qi, E. Drabek, and D. Liu, "Evaluation of Text Summarization in a Cross-lingual Information Retrieval Framework," Center for Language and Speech Processing, Johns Hopkins University, Baltimore, MD, Tech. Rep., June 2002.

[9] G. A. Miller, "Wordnet: a lexical database for english," in *HLT '91: Proceedings of the workshop on Speech and Natural Language*. Morristown, NJ, USA: Association for Computational Linguistics, 1992, pp. 483–483.

[10] N. Verma and P. Bhattacharyya, "Automatic lexicon generation through wordnet," in *International Conference on Global Wordnet (GWC 04)*, 2004.

[11] G. Ramakrishnan and P. Bhattacharyya, "Text representation with wordnet synsets," in *Special Issue on the Application of Natural Language to Information Systems*, 2004.

[12] T. J. Ross, *Fuzzy Logic With Engineering Applications*. McGRAW-HILL Internationsl Editions, 1997.

[13] M. Frank, "On the simultaneous associativity of f(x,y) and x+y-f(x,y)," *Aequationes Math*, vol. 19, pp. 194–226, 1979.