Developing Learning Strategies for Topic-based Summarization

You Ouyang
Deparment of Computing
Hong Kong Polytechnic University
superoil1983@hotmail.com

Sujian Li
Institute of Computational Linguistics
Peking University
lisujian@pku.edu.cn

Wenjie Li
Deparment of Computing
Hong Kong Polytechnic University
cswjli@comp.polyu.edu.hk

ABSTRACT

Most up-to-date well-behaved topic-based summarization systems are built upon the extractive framework. They score the sentences based on the associated features by manually assigning or experimentally tuning the weights of the features. In this paper, we discuss how to develop learning strategies in order to obtain the optimal feature weights automatically, which can be used for assigning a sound score to a sentence characterized with a set of features. The two fundamental issues are about training data and learning models. To save the costly manual annotation time and effort, we construct the training data by labeling the sentence with a "true" score calculated according to human summaries. The Support Vector Regression (SVR) model is then used to learn how to relate the "true" score of the sentence to its features. Once the relations have been mathematically modeled, SVR is able to predict the "estimated" score for any given sentence. The evaluations by ROUGE-2 criterion on DUC 2006 and DUC 2005 document sets demonstrate the competitiveness and the adaptability of the proposed approaches.

Categories and Subject Descriptors

I.7.5 [**Document and Text Processing**]: Document Capture – *document analysis*.

General Terms

Algorithms, Performance, Design, Experimentation

Keywords

Document Summarization, Support Vector Regression

1. INTRODUCTION

Currently, most successful summarization systems are implemented by extracting the most salient sentences from the given documents into the summary. For these systems, sentence scoring that determines which sentences are more important than others is a key process for ranking and selecting sentences. Typically, the scoring methods calculate the combinational effects of various features which are designed to characterize the different aspects of the sentences and/or their relevance to the topics.

Certainly, the selection of the appropriate features highly influences system performance. That is why many high performance systems have devoted much effort to exploring effective features. However, the combination of the features should also be an equally important issue. Yet so far not much attention has been paid to it. Normally, the features are simply combined by a linear function in which the weights are assigned manually or tuned experimentally. There are several shortcomings with these methods. (1) The performance of manually assigned weights is not predictable. (2) When the feature set becomes large, the complexity of experimentally tuning weights grows exponentially. Our objective in this paper is to explore how the optimal weights can be obtained automatically by developing learning strategies. This involves two fundamental issues which must be addressed by any learning based approach, i.e. learning models and training data.

In the past, machine learning approaches have been successfully applied in many natural language processing, information extraction and question answering tasks and so on. However, they have not been well acknowledged in extractive summarization. The bottleneck is the lack of appropriate and sufficient training data which is necessary for training models. While a few learning based systems attempt to solve this problem by utilizing the existing key sentence sets or the document classification information existing on the WWW, most systems are still in the stage of using human assigned or tuned weights.

In this paper, we apply a machine learning approach to topic-based summarization by regarding sentence scoring as a regression problem. The regression function is learned from the Support Vector Regression (SVR) model, which is the regression type of Support Vector Machine (SVM) and is capable of building state-of-art optimum approximation functions. It provides a way of combining the features automatically and effectively. To save the costly manual annotation time and effort, we construct training data automatically from the document sets where the reference summaries generated by human have been provided. To make use of human summaries, we develop *N*-gram methods to approximately measure the "true" sentence scores. The SVR models then learn how to relate the "true" scores of the sentences to the corresponding features. Once the relations have been mathematically modeled, they are capable of predicting the "estimated" score for any given sentence. In experiment the

CIKM'07, November 6–8, 2007, Lisboa, Portugal.

performance of the *N*-gram methods is compared to show how to find the best method for training data construction.

The remainder of the paper is organized as follows. Section 2 briefly introduces the related work. Section 3 details the training data construction methods and SVR learning models. Section 4 then presents experiments and evaluations as well as discussions. Section 5 finally concludes the paper.

2. RELATED WORK

The most straightforward way to apply machine learning approaches in text summarization is to regard the sentence extraction task as a binary classification problem. Kupiec et al [11] developed a trainable summarization system which adopts various features and uses a Bayesian classifier to learn tuning the feature weights according to a set of documents with the corresponding extracts, and the system performed better than any other system using only a single feature. From a set of documents where sentences have been annotated manually, Hirao et al [18] trained a SVM model to learn how to extract the important sentences from the given documents. The model outperformed other learning based models such as those using decision-tree or boosting methods in the Japanese Text Summarization Challenge (TSC). Later, Zhou and Hovy [5] introduced a HMM-based model to estimate the extract desirability of an English sentence and trained the parameters on data collected from YAHOO. The resultant system was comparable to the best system in DUC 2001. Following the same idea, Zhao, Wu and Huang [13] applied the Conditional Maximum Entropy model in DUC 2005 topic-based summarization task. Unfortunately, only moderate performance was achieved. Since classification based models require training corpus with sufficient size and high quality, they are largely limited by the costly annotation time and efforts, and thus so far are not widely acknowledged.

Recently, statistical analysis has been concerned with examining human generated summaries. Several sentence scoring methods based on human summaries and simple statistic measures have yielded powerful summarization systems. For example, Nenkova and Vanderwende [1] examined the impact of word frequency on summarization by extensively studying the relations of term frequency and human practice in summarization. It is remarkable that the system simply based on frequency features could perform unexpectedly well in MSE 2005. Conroy and Schlesinger [10] also defined "oracle" score which was calculated from the probability distribution of Uni-grams in human summaries. By extracting sentences depending on the "oracle" scores, they built a system which performed even better than some human generated summaries in DUC 2006. We believe that the idea of comparing sentences with human summaries should yield a good scoring method for ranking and selecting sentences. It should be able to give sentences the credible scores and provide a good mean to automatically generate training data for regression models, which are more similar to the sentence scoring task than classification models.

3. TOPIC-BASED SUMMARIZATION

3.1 Background

The task of DUC topic-based summarization requires creating from a set of relevant documents (most set contains 25 documents) a brief, well-organized and fluent summary to the information seeking need which is indicated in the topic description. Here is an example of the topic description.

<topic>

<num> D0601A </num>

<title> Native American Reservation System - pros and cons </title>

<narr>

Discuss conditions on American Indian reservations or among Native American communities. Include the benefits and drawbacks of the reservation system. Include legal privileges and problems.

</narr>

</topic>

For each topic, four human summarizers are asked to provide a 250-word summary of the topic from the 25 related documents for automatic evaluation.

NIST assessors developed a total of 50 DUC topics both in DUC2005 data set and DUC2006 data set. The topic structures of the two data sets are almost the same except that the source and number of documents are somewhat different. Each DUC2005 topic includes 25-50 related documents selected from the *Los Angeles Times* and *Financial Times of London*, while each DUC2006 topic is composed of exactly 25 documents from *Associated Press*, *New York Times* and *Xinhua* Newswire.

3.2 Feature Design

Sentences are scored according to the features. Therefore, features play an important role in our work. We design a set of features containing three topic dependent features and four topic independent features to characterize the sentences for topic-based summarization task. The design criterion we follow is: we try to capture the important information that a sentence conveys in a more direct and explicit way. The feature values are calculated as follows.

(1) Word Matching Feature

$$f_{word}(s) = \sum_{t_j \in s} \sum_{t_i \in q} same(t_i, t_j)$$

where f is the feature value, q is the topic description. The function $same(t_i, t_j) = 1$ if $t_i = t_j$, and 0 otherwise.

(2) Semantics Matching Feature

$$f_{wordnet}(s) = \sum_{t_j \in s} \sum_{t_i \in q} similarity(t_i, t_j)$$

where the function $similarity(t_i, t_j)$ is the lesk similarity function introduced in [14], which is WordNet-based and scales the semantic relation between two words.

(3) Name Entity Matching Feature

$$f_{entity}(s) = |\, entity(s) \cap entity(q) \,|$$

where $|\, entity(s) \cap entity(q) \,|$ is the number of the named entities in both *s* and *q*. A named entity is usually defined as a word, or a sequence of words that can be classified as a person name, organization, location, date, time, percentage or quantity. Here only four classes (person, organization, location, date) are involved,

(4) Document Centroid Feature

$$f_{centroid}(s) = \sum_{t_j \in s} tfidf(t_j)$$

where $tfidf(t_j)$ is the *tf-idf* score of t_j in the whole data set.

(5) Named Entity Number Feature

$$f_{entityno}(s) = |\, entity(s) \,|$$

where $|\, entity(s) \,|$ is the number of named entities in *s*.

(6) Stop Word Penalty Feature

$$f_{stopword}(s) = |\, stopword(s) \,|$$

where $|\, stopword(s) \,|$ is the number of the stop words in *s*.

(7) Sentence Position Feature

$$f_{position}(s) = 1 - \frac{i-1}{n}$$

where *n* is the total number of the sentences and *s* is the *i*th sentence in the document.

3.3 Training Data Construction: *N*-Gram Methods based on Human Summaries

3.3.1 Overview

To construct training data, we propose several *N*-gram methods to assign "true" sentence scores with reference to human summaries. The main hypothesis behind the use of them is: if the human summaries are the excellent sum-ups of the documents which contain abundant information, the sentences in the documents which are more similar to those in human summaries should be more likely to be the good sum-ups as well, and thus they should stand a good chance to be assigned higher scores by the scoring methods.

Given a document set *D* and a human summary set $H = \{H_1, \cdots, H_m\}$ (H_i is the human summary generated by the *i*th linguist), each sentence *s* in *D* is assigned a score $score(s \mid H)$. Basically, the scoring methods compute the *N*-gram (more specific, Uni-gram or Bi-gram) probabilities of *s* to be recognized as a summary sentence given human summaries.

3.3.2 Frequency-based Methods

The probability of an *N*-gram *t* under a single human summary H_i can be calculated by

$$p(t \mid H_i) = freq(t) / |\, H_i \,|$$

where $freq(t)$ is the frequency of *t* and $|\, H_i \,|$ is the number of the words in H_i. To obtain the probability of *t* under all human summaries, we propose two strategies, namely *Maximum* and *Average* strategies. *Maximum* emphasizes on the diversity of different summaries. It selects the largest probability under single human summaries,

$$p_{Max}(t \mid H) = \underset{H_i \in H}{Max}(p(t \mid H_i))$$

In contrast, *Average* treats all human summaries as a whole and computes probability average,

$$p_{Avg}(t \mid H) = \frac{1}{|H|} \sum_{H_i \in H} p(t \mid H_i)$$

The overall score of a sentence *s* is then the sum of the probabilities of all the *N*-grams it contains, indicated by $t_j \in s$,

$$score(s \mid H) = \sum_{t_j \in s} p(t_j \mid H)$$

Considering the fact that all human summaries are almost of the same length, the equation of computing probability of *N*-gram *t* under a single human summary H_i can be simplified to

$$p(t \mid H_i) = freq(t)$$

Finally, we have two alternative sentence scoring methods based on *N*-gram frequencies, i.e.

$$score_{Max}(s \mid H) = \sum_{t_j \in S} \underset{H_i \in H}{Max}(freq(t_j))$$

and

$$score_{Avg}(s \mid H) = \sum_{t_j \in S} \sum_{H_i \in H} freq(t_j)$$

3.3.3 Appearance-based Methods

In addition to frequency, binary *N*-gram appearance judgment can also be applied in sentence score calculation. Based on *N*-gram appearance, the probability of an *N*-gram *t* under single human summary is

$$p'(t \mid H_i) = \begin{cases} 1, & t \text{ appears} \\ 0, & others \end{cases}$$

Accordingly, sentence scoring methods are revised as

$$score_{Max}(s \mid H) = \sum_{t_j \in S} \underset{H_i \in H}{Max}(p'(t_j \mid H_i))$$

and

$$score_{Avg}(s \mid H) = \sum_{t_j \in S} \sum_{H_i \in H} p'(t_j \mid H_i)$$

3.3.4 Remark

Though the methods introduced above are quite simple, they are considerably effective in picking up good summary sentences. You will see in Experiment 1 described in Section 4.2, the *N*-gram-based methods that rely on human summaries can produce the extractive summaries that are even better than the human

summaries. However, human summaries are normally unavailable when document sentences are scored for inclusion in a machine generated summary. Therefore, the *N*-gram methods cannot be applied to the practical summarization task directly. One has to seek for alternative ways to approximate sentence scoring process. Our solution is to build a regression based approximation function *f*. The input to the function will be a set of features $F(s)$ that we select to characterize any given sentence *s*. The output from it will be the "estimated" score $\hat{score}(s)$ of *s*. Though the *N*-gram methods cannot be applied directly, they can be used to construct the training data which is then used to learn the core component of our solution, i.e. the regression function.

3.4 Model Learning: SVR-based Methods

Under feature-based summarization framework, normally the scoring function needs to combine the impacts of the features. A common way is to use the linear combination of the features by tuning the weights of the features manually or experimentally. A problem of such method is that when the number of the features gets larger, the complexity of assigning weights grows exponentially. Some attempts have been made on using classification models to tune the weights automatically to avoid the exponential complexity. However, scoring function is a continuous real-value function while classification models are usually adopted to solve discrete problems, thus it is imprecise to use classification models on the sentence scoring task. In this section, we explore regression model for sentence scoring such that credible and controllable solutions can be expected.

Models are trained from the document sets *D* where the human summaries *H* are given. Each *s* in *D* associates with a score $score(s \mid H)$ and a feature vector $F(s)$. We obtain the training data $\{(score(s \mid H), F(s)) \mid s \in D\}$ by correlating the sentence's score and its feature set together. Thus the task of predicting the score of the sentence *s* in another document sets *D'* given its $F(s)$ is just a regression problem, to generate the regression function $f : F(s) \rightarrow \hat{score}(s)$ based on $\{(score(s \mid H), F(s)) \mid s \in D\}$.

The linear *v*-SVR model [15] is used to learn the regression function. It chooses the optimum function $f_0(x) = w_0 \cdot x + b_0$ from the candidate function set $\{w \cdot x + b \mid w \in R^n, b \in R\}$ by minimizing the structure risk function

$$\Phi(w, b, \varepsilon) = \frac{1}{2} \| w \|^2 +$$

$$C\left(\frac{1}{|D|} \sum_{s_i \in D} L(score(s_i) - (w \cdot F(s_i) + b)) + v\varepsilon\right)$$

where $L(x)$ is the ε-insensitive lost function defined as $L(x)=|x|-\varepsilon$, if $x>\varepsilon$ and $L(x)=0$, otherwise. *C* and *v* are the weights to balance the factors. Comparing to traditional regression models, SVR is more generative and robust by introducing a normalization factor $\frac{1}{2} \| w \|^2$ in the risk function.

Once the regression function f_0 is learned, we define

$$\hat{score}(s) = f_0(F(s)) = w_0 \cdot F(s) + b_0$$

Normally, a summary is limited in length to a maximum number of words. With this consideration, the score should be normalized by the sentence length. The score function is then refined as

$$\hat{score}_{norm}(s) = \frac{1}{|s|} \hat{score}(s)$$

where $|s|$ is the number of the words in *s*.

3.5 Redundancy Removal

Under the above sentence scoring approach, if the terms of two sentences are very similar, the sentences may probably have approximate feature values, therefore, they may also probably have approximate scores. Thus the extracted summary may include high score sentences which are very similar, this will cause redundant information in summary. To solve this problem, a maximum marginal relevance (MMR) approach is applied during the sentence selection process. First all the sentences are ordered by score from highest to lowest, and then the summary sentences are selected iteratively, each time the current candidate sentence is compared to the sentences already in the summary. If the sentence is not too similar to any sentence already in the summary (the similarity value of the two sentences is lower than a given threshold), the sentence is then selected to the summary. The iteration is repeated until the length of the summary reach the upper limit. In this paper, we use the cosine metric as the similarity function, and the threshold is set to 0.6.

4. EXPERIMENT AND EVALUATION

4.1 Experiment Set-up

We set up our preliminary experiments on DUC 2006 and then further test on DUC 2005 document sets. All documents are pre-processed by removing stop words and conducting stemming. Named entities including person and organization names, locations and time are automatically tagged by GATE [1]. According to the task definition system generated summaries are strictly limited to 250 English words in length. After sentence scoring, we select the highest scored sentences from the original documents into the summary until the word (actually the sentence) limitation is reached. Considering the focus of this study, no post-processing such as sentence compression or information fusion is carried out. We will present the results of the eight combinations with consideration of *N*-grams (Uni-gram or Bi-gram), probability calculations (frequency or appearance) and scoring strategies (Maximum or Average).

4.2 Evaluation Metrics Introduction

In DUC2005 and DUC2006, summaries are evaluated by several manual or automatic evaluation metrics [8][9]. In this paper, we use two of the DUC evaluation criteria, ROUGE-2 and ROUGE-SU4, to compare our systems built upon the proposed scoring and learning methods with human summarizers and several top performing DUC systems. ROUGE (Recall Oriented Understudy for Gisting Evaluation) [3] is a state-of-art automatic summarization evaluation method based upon *n*-gram comparison.

[1] It is publicly available from http://gate.ac.uk/.

In DUC, NIST assessors developed several model summaries manually for each DUC topic. Using model human summaries as the golden standard, ROUGE evaluates summaries submitted by comparing them with the model summaries. For example, ROUGE-2 evaluates a system summary by matching its bigrams against the model summaries:

$$R_n(S) = \frac{\sum_{j=1}^{h} \sum_{t_i \in S} Count(t_i | S, H_j)}{\sum_{j=1}^{h} Count(t_i | H_j)}$$

where S is a summary to be evaluated, $H_j (j = 1,...,h)$ is h model human summaries, t_i is a bigram in S. $Count(t_i | H_j)$ is the number of times the bigram t_i occurred in the j-th model human summary H_j and $Count(t_i | S, H_j)$ is the number of times t_i occurred both in summary S and H_j.

ROUGE-SU4 is very similar to ROUGE-2, the difference is that it matches unigrams and skip-bigrams of a summary against model summaries instead of bigrams. A skip-bigram is a pair of words in their sentence order, allowing for gaps within a limited size.

Though ROUGE is just based on simple *n*-gram matching framework, it has been working well in DUC. In DUC2005, ROUGE-2 had a Spearman correlation of 0.95 and a Pearson correlation of 0.97 compared with human evaluation.

4.3 Experiment 1: Evaluation of *N*-Gram based Methods on DUC 2006

The aim of the first set of experiments is to prove the rationality of *N*-gram methods in assigning the "true" sentence score by using the score directly for selecting sentences. Table 1 shows the average ROUGE-2 value and the 95% confidential interval (CI) over 50 document sets for the systems based upon the *N*-gram methods (in grey shade) and the best summary created by human. Based on this evaluation, the systems developed with *N*-gram methods even achieve much better performance than human summarizer, showing that the score assigned by *N*-gram methods is reliable. Most important, it demonstrates that the documents themselves contain the sentences that are good enough to produce a summary even better than human summaries when evaluated by ROUGE-2, which shows the potential of extractive summarization.

4.4 Experiment 2: Evaluation of Regression Models on DUC 2006

The *N*-gram methods are designed particularly for constructing training data for SVR model learning. Now, we conduct the second set of experiments to examine how well the training data constructed can be used for learning regression functions. In this set of experiments, 50 document sets are divided into training data composed of 5 sets and test data composed of 45 sets. Cross-validation is applied to leverage the results. In this paper, we use LIBSVM [2] to implement the *v*-SVR model and set the parameters of LIBSVM as default values. Table 2 presents the results of the SVR-based systems (in grey shade) together with the 8 top performing systems at DUC 2006 (labeled as 24, … 2), one worst human summary (labeled as A), and a baseline system which uses the linear function to combine the features. The weights of the features in the baseline system are set manually. The average ROUGE-1, ROUGE-2 and ROUGE-SU4 value and corresponding 95% confidential intervals of all systems are all proposed.

Table 1. Results of Sentence Scoring Methods given Human Summaries on DUC 2006

Submission	Average Rouge-2 and CI	Submission	Average Rouge-2 and CI
Bi+Appr+Max	**0.1711 (0.1608, 0.1830)**	Uni+Appr+Avg	0.1468 (0.1358, 0.1584)
Bi+Appr+Avg	0.1666 (0.1563, 0.1772)	Best human Summary	0.1326 (0.1160, 0.1520)
Bi+Freq+Max	0.1603 (0.1489, 0.1726)	Uni+Freq+Max	0.1160 (0.1071, 0.1260)
Uni+Appr+Max	0.1603 (0.1503, 0.1710)	Uni+Freq+Avg	0.1149 (0.1058, 0.1245)
Bi+Freq+Avg	0.1578 (0.1427, 0.1672)		

Table 2. Results of Sentence Scoring Methods based on SVR Model in DUC 2006

Submission	Average Rouge-1 (CI)	**Average Rouge-2 (CI)**	Average Rouge-SU4(CI)
A	0.4582 (0.4496, 0.4682)	0.1036 (0.0926, 0.1162)	0.1683 (0.1604, 0.1773)
24	0.4111 (0.4049, 0.4171)	0.0956 (0.0914, 0.0998)	0.1553 (0.1513, 0.1591)
Uni+Freq+Max	**0.4018 (0.3959, 0.4078)**	**0.0926 (0.0883, 0.0969)**	**0.1485 (0.1443, 0.1525)**
15	0.4028 (0.3965, 0.4084)	0.0910 (0.0867, 0.0948)	0.1473 (0.1437, 0.1507)
Uni+Freq+Avg	0.3986 (0.3925, 0.4045)	0.0906 (0.0862, 0.0952)	0.1464 (0.1423, 0.1502)
12	0.4049 (0.3992, 0.4105)	0.0899 (0.0858, 0.0939)	0.1476 (0.1436, 0.1514)
Uni+Appr+Avg	0.3977 (0.3919, 0.4030)	0.0899 (0.0860, 0.0937)	0.1455 (0.1418, 0.1489)
8	0.4001 (0.3937, 0.4056)	0.0895 (0.0854, 0.0934)	0.1461 (0.1425, 0.1494)

Uni+Appr+Max	0.3976 (0.3920, 0.4030)	0.0895 (0.0856, 0.0931)	0.1454 (0.1417, 0.1487)
Bi+Appr+Avg	0.3990 (0.3930, 0.4046)	0.0893 (0.0853, 0.0937)	0.1449 (0.1410, 0.1487)
Bi+Freq+Max	0.3985 (0.3926, 0.4040)	0.0889 (0.0849, 0.0935)	0.1450 (0.1411 0.1488)
Bi+Freq+Avg	0.3973 (0.3914, 0.4028)	0.0888 (0.0847, 0.0934)	0.1444 (0.1404, 0.1483)
23	0.4044 (0.3982, 0.4097)	0.0879 (0.0837, 0.0920)	0.1449 (0.1410, 0.1485)
Bi+Appr+Max	0.3947 (0.3894, 0.4001)	0.0876 (0.0838, 0.0917)	0.1429 (0.1391, 0.1463)
28	0.3992 (0.3936, 0.4046)	0.0870 (0.0833, 0.0910)	0.1452 (0.1416, 0.1488)
Baseline	0.3744 (0.3668, 0.3813)	0.0751 (0.0709, 0.0792)	0.1299 (0.1257, 0.1340)

Table 3. Results of Sentence Scoring Methods based on SVR Model in DUC 2005

Submission	Average Rouge-2	Submission	Average Rouge-2
…	…	Bi+Appr+Avg	0.0720(0.0684, 0.0754)
H	0.0897	17	0.0717
Uni+Freq+Max	**0.0757 (0.0720, 0.0791)**	Bi+Freq+Avg	0.0713(0.0681, 0.0745)
Uni+Freq+Avg	0.0747 (0.0711, 0.0783)	Bi+Freq+Max	0.0709 (0.0675, 0.0741)
Uni+Appr+Avg	0.0726 (0.0691, 0.0759)	8	0.06960
15	0.0725	4	0.06860
Bi+Appr+Max	0.0724 (0.0689, 0.0761)	…	…
Uni+Appr+Max	0.0720 (0.0684, 0.0755)	Baseline	0.0631 (0.0594, 0.0668)

Table 5: Result of combining different features with Uni+Freq+Max ("√" means the feature is used)

$f_{centroid}$	f_{word}	$f_{position}$	$f_{stopword}$	$f_{entity} + f_{entityno}$	$f_{wordnet}$	Average Rouge-2 and CI
√						0.06030 (0.05711, 0.06354)
	√					0.06280 (0.05981, 0.06563)
	√	√				0.06407 (0.06117, 0.06698)
√	√					0.07056 (0.06731, 0.07376)
√	√	√				0.07088 (0.06753, 0.07404)
√	√		√			0.07286 (0.06944, 0.07612)
√	√	√	√			0.07467 (0.07109, 0.07812)
√	√	√	√	√		0.07509 (0.07150, 0.07857)
√	√	√	√	√	√	**0.07556 (0.07201, 0.07912)**

The SVR-based systems perform comparably to the top performed systems at DUC 2006. All of them outperform the baseline system. These results clearly show the advantage benefited from applying SVR in combining features and the rationality of generating training data using the *N*-gram methods based on human summaries. Note that this time "Bi+Appr+Max" falls off significantly but "Uni+Freq+Max" climbs up.

4.5 Experiment 3: Evaluation of Regression Models on DUC 2005

To further examine the capability of the SVR-based score approximation methods, we extend Experiment 2. The models are trained again on 5 DUC 2006 document sets, while the evaluation runs against the 50 document sets of DUC 2005. The results are presented in Table 3. *H* represents the result when evaluating the worst human summaries The ROUGE-2 values are reported. Since the confidential interval of the systems is not given officially at DUC 2005 [8], only the average scores are listed.

It is quite encouraging that the best SVR-based system performs better than all the systems at DUC 2005. Even for the worst SVR-based system, it can be ranked in the third place. The results verify the reliability and the adaptability of the SVR-based learning approaches to a certain extent, though a large scale of experiments are still needed to further confirm this conclusion.

Table 4: Results of manually assigned weights in DUC 2005

Human Assigned Weight 1	0.0631 (0.0594, 0.0668)
Human Assigned Weight 2	0.0572 (0.0537, 0.0606)
Human Assigned Weight 3	0.0645 (0.0606, 0.0683)
Human Assigned Weight 4	0.0657 (0.0623, 0.0691)
Human Assigned Weight 5	0.0620 (0.0586, 0.0651)
Average of 5 systems	0.0625

4.6 Experiment 4: Evaluation of More Manually Assigned Weights on DUC 2005

The weights of the baseline system in Experiment 3 are assigned manually. One may argue that these weights may happen to produce the worst ROUGE-2 scores. To clarify this concern, we conduct Experiment 4 with five different sets of weights assigned by human, and present the results in Table 4. *Human assigned weights 1* represents the one used as the baseline system in Experiment 3, which is already above the average in this experiment.

4.7 Experiment 5: Evaluation of Different Feature Sets on DUC 2005

As we know, how the feature set is formed directly influences the performance of SVR-based systems. In this set of experiments, we fix on "Uni+ Freq+ Max", which is the best scoring method in Experiment 3. The seven features introduced in Section 3.2 are selected and combined gradually to form nine different feature sets for learning functions. The evaluation is again carried out on DUC 2005 document sets. Table 5 illustrates how the performance is improved when more features are included. When more appropriate features are involved, the SVR models are capable of tuning the weights of the incremented feature sets such that the optimal combinations could always be achieved. Thus a more accurate score approximation function can be obtained. The result proves that SVR is effective in searching for optimal weights.

4.8 General Discussion

Comparing Uni-gram and Bi-gram methods, we find that the Bi-grams spread sparser than Uni-grams in the document sets. The number of sentences which receives a zero score in Bi-gram-based methods (which means the Bi-grams of these sentences never appear in human summaries) is much larger than the number in Uni-gram-based methods. It is about 75% vs. 20%. Thus the data sparsity problem is more serious in Bi-gram methods than in Uni-gram methods. It unavoidably influences the performance of machine learning approaches.

Comparing frequency-based and appearance-based methods, we can see that frequency-based methods perform better even though not very significantly. There may be two reasons to explain it. First, frequency-based methods maintain more information of the original documents. Second, most features designed are also based upon frequency calculation.

5. CONCLUSION

This paper proposes the methods for constructing training data based on human summaries and training sentence scoring models based on regression models. The results are significant. Summaries extracted based on these methods can perform as well as human-generated abstracts when evaluated by ROUGE-2 on DUC 2005 and 2006 document sets. More important, compared to the other systems in DUC competitions, our SVR-based system can achieve very good performances with very simple work on sentence compression and redundant removal.

6. ACKNOWLEDGMENTS

The work described in this paper was partially supported by Hong Kong RGC Project (No. PolyU5211/05E) and partially supported by china NSFC Project (No. 60603093) and IBM-PKU Joint Research Project.

7. REFERENCES

[1] Ani Nenkova and Lucy Vanderwende. *The Impact of Frequency on Summarization.*.MSR-TR-2005-101. Microsoft Research Technical Report, 2005.

[2] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: A Library for Support Vector Machines*, 2001. http://www.csie.ntu.edu.tw/~cjlin/libsvm

[3] Chin-Yew Lin and Eduard Hovy. *Manual and Automatic Evaluation of Summaries*. In Document Understanding Conference 2002 http:/duc.nist.gov, 2002.

[4] Christiane Fellbaum, editer. WordNet: *An Electronic Lexical Database*. The MIT Press, Cambridge London, 1998.

[5] Deepak Ravichandran, Eduard Hovy. *Learning Surface Text Patterns for a Question Answering System*. In Proceedings of the 40th Annual Meeting of the ACL, pages 41-47, 2002.

[6] Dragomir R. Radev, Jahna Otterbacher, Hong Qi, Daniel Tam. *MEAD ReDUCs: Michigan at DUC 2003*. In Document Understanding Conference 2003, 2003. http://duc.nist.gov

[7] Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan. *GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications*. In Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics, pages 168-175, 2002.

[8] Hoa Trang Dang. *Overview of DUC 2005.* Document Understanding Conference 2005 http://duc.nist.gov, 2005.

[9] Hoa Trang Dang. *Overview of DUC 2006*. Document Understanding Conference 2006 http://duc.nist.gov, 2006.

[10] John M.Conroy, Judith D.Schlesinger, Dianne P.O'Leary. *Topic-Focused Multi-document Summarization Using an Approximate Oracle Score*. In Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, pages 152-159, 2006.

[11] Julian M. Kupiec, Jan Pedersen, and Francine Chen. *A Trainable Document Summarizer*. In Edward A. Fox, Peter Ingwersen, and Raya Fidel, editors, Proceedings of the 18th Annual International ACM SIGIR Conference on Research

and Development in Information Retrieval, pages 68-73, 1995.

[12] Liang Zhou and Eduard Hovy. *A Web-trained Extraction Summarization System*. In Proceedings of HLT-NAACL 2003, pages 205-211, 2003.

[13] Lin Zhao, Lide Wu, Xuanjing Huang. *Fudan University at DUC 2005.* In Document Understanding Conference 2005. http://duc.nist.gov, 2005.

[14] Satanjeev Banerjee, Ted Pedersen. *An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet*. In Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing (CICLING-02). pages 136-145, 2002.

[15] Schölkopf, B., Bartlett, P. L., Smola, A., & Williamson, R.C.*Support Vector Regression with Automatic Accuracy Control.* In Proceedings of the 8th International Conference on Artificial Neural Networks, pages 111-116, 1998.

[16] Seeger Fisher, Brian Roark. *Query-Focused Summarization By Supervised Sentence Ranking and Skewed Word Distributions.* In Document Understanding Conference 2006. http:/duc.nist.gov, 2006.

[17] Steve R. Gunn. *Support Vector Machines for Classification and Regression*, Technical Report, Image Speech and Intelligent Systems Research Group, University of Southampton, 1998.

[18] Tsutomu Hirao, Hideki Isozaki. *Extracting Important Sentences with Support Vector Machines*. Proceedings of the 19th International Conference on Computational Linguistics, pages 342-348, 2002.

[19] Vladimir Vapnik. *Statistical Learning Theory*. John Wiley and Sons, New York, 1998.