An MAP Based SentenceRanking Approach to Automatic Summarization

Xiaofeng Wu
National Laboratory of Pattern Recognition
Institute of Automation, Academy of Sciences
Beijing, China
xfwu@nlpr.ia.ac.cn

Chengqing Zong
National Laboratory of Pattern Recognition
Institute of Automation, Academy of Sciences
Beijing, China
cqzong@nlpr.ia.ac.cn

Abstract:

While the current main stream of automatic summarization is to extract sentences, that is, to use various machine learning methods to give each sentence of a document a score and get the highest sentences according to a ratio. This is quite similar to the current more and more active field -- learning to rank. A few pair-wised learning to rank approaches have been tested for query summarization. In this paper we are the pioneers to use a new general summarization approach based on learning to rank approach， and adopt a list-wised optimizing object MAP to extract sentences from documents, which is a widely used evaluation measure in information retrieval (IR). Specifically, we use SVMMAP toolkit which can give global optimal solution to train and score each sentences. Our experiment results shows that our approach could outperform the stand-of-the-art pair-wised approach greatly by using the same features, and even slightly better then the reported best result which based on sequence labeling approach CRF.

Keywords:

Summarization; MAP

1. Introduction

After the first few decays of slow progress since Luhn[1], document summarization suddenly became a rapidly evolving subfield of Information Retrieval (IR) due to the booming of the internet. A summary can be loosely defined as a text that is produced from one or more texts and conveys important information of the original text(s), usually it is no longer than the half of the original text(s) or, significantly less[2]. In summarization evaluation, the most famous one should be Document Understanding Conference DUC "http://duc.nist.gov", which has moved to Text Analysis Conference TAC "www.nist.gov/tac" since 2007. DUC published some summarization corpus every year and encouraged researchers to compete. It is obvious that, in the age of information explosion, document summarization will be greatly helpful to the internet users; besides, the used techniques could also find their applications in speech techniques and multimedia document retrieval, etc.

978-1-4244-6899-7/10/$26.00 ©2010 IEEE

There are several taxonomies of summarization such as: indicative, informative and evaluative, according to their functionality; single-document and multi-document, according to the amount of the input documents; generic and query-oriented , according to applications; extractive and abstractive according to how to use the original text ; supervised and unsupervised according to which machine learning algorithm is used[3].

Extractive summarization is still attracting a lot of researchers [4-6] and many practical systems, say, MEAD "http://www.summarization.com/mead/", have been produced. Using supervised or unsupervised machine learning algorithms to extract sentences is currently the mainstream of the extractive summarization. In this paper we focus on extractive supervised generic single-document summarization.

Among the various machine learning methods in sentence extraction, we choose Learning to Rank(LTK)[7-8] scheme. Because first, LTK has become a very active area in IR; second, using LTK is to let sentences within a document compare with themselves, rather than to train a model and let sentences 'compete' among documents, as the traditional methods do. This 'intra or inter' point of view is quite interesting and has seldom been studied.

Our main contributions are: First, we use a list-wise LTK scheme: SVMMAP[9] for sentence selection based summarization, it is the first time this optimal scheme has been used in summarization; Second, previous works have only studied the ability of LTK schemes in query-summarization, it is the first time this method has been used in general-summarization; Third, we make a primitive comparison of several popular ranking schemes, list-wised or pair-wised, in summarization.

The remainder of this paper is organized as follows: in Section 2 we give related work; our motivation is described in Section 3; in Section 4 we give experiment setup and results; in the last Section we draw conclusions and discuss the future work.

2. Related work

2.1. *Intra vs. Inter*

Nowadays most of supervised extractive methods deem summarization as a classification problem, and focus on finding good machine learning algorithms that can properly combine the features like proper names, sentence positions and sentence similarity etc. Various algorithms have been tried including: Beyesian classifier [10], decision tree [11], HMM [12], ME[13], CRF[14], Semi-CRF[6], genetic algorithm[4], etc.

According to Wang[15], all these approaches share the same assumption that all sentences from various documents should be comparable with respect to the classify information. That is to compare sentences according to a model trained upon all documents, which we call it '*Inter*'. Generally, the classes used in the summary problem are: the sentence IS a summary or NOT. The summary classify problem is that on every sentence (represented by its features) a score should be computed based upon each class, and the class with the higher score will be chosen. If dealing with similar (the 'similar' means writing style, and vocabulary et al.) documents, this assumption is hold. But when dealing with various documents come from different fields, this assumption is obviously inappropriate.

Therefore, Wang[15] tried to use LTK in summarization problem. The author claimed that using ranking scheme in summarization is to compare each sentence within a document, thus might hurt less by the dissimilarity of documents, which is the shortcoming of summarization under classification point of view. The author claimed that using Ranking SVM[8] was a good choice for summarization. Under this framework, the score of sentences is computed by comparing each other within the same document. This we call it '*Intra'*.

Other than Ranking SVM, Metzler[16] investigated the ability of a few more LTk schemes, such as, SVR and gradient boosted decision tree(GBDT)[17] in query-based summarization.

Both Wang and Metzler studied only query-based summarization, but they did not tried this approach in generic one. Anther deficit is that they only investigated pair-wised LTK scheme, not the list-wised approaches, which are more powerful and novel.

2.2. *List-wise vs. Pair-wise*

LTK algorithms can fall into three categories: point-wise, pair-wise and list-wise. The point-wise approach uses regression or classification on single object to do the ranking problem[18], while the pair-wise approach do it on the object pairs[8]. Both of them are using existing regression or classification algorithms, but model the ranking problem indirectly. The list-wise approach, on the contrary, is designed directly according to the ranking problem[19-20]. In this work we consider only pair-wise and list-wise approaches.

In both information retrieval and machine learning, how to learn a model in training data and then sort objects according to their degree of relevance, preference or importance, that is, the problem of LTK has been an active and growing area recently. Set an optimal target such as nDCG, precision at K, MAP, people employ various algorithms to adapt to the target, which include: logistic regression, ranking SVM (SVMs), neural network, and perception. Several useful ranking algorithms have been published like Burges et al.'s RankNet[21], Freund et al.'s RankBoost[22], and Herbrich et al.'s Ranking SVM[23]. Yet all these algorithms are '*pair-wised*'. Pair-wised approaches are defined as 'the learning task is formalized as classification of object pairs into two categories (correctly ranked or not). There were many applications in IR of this ranking scheme, such as Joachims[8] and Burges [21].

The main advantage of using Pair-wised approaches is obvious that the existing classification methods are ready to apply in this framework. Nevertheless, its main disadvantage of using it is also obvious that rather than to minimize errors in ranking, pair-wised approaches' objective is to minimize the errors in document or other unit's pairs.

Cao[19] proposed another ranking scheme called ListNet, which employs document lists as the learning instances instead of document pairs. Cao refer to his method as ListNet, and he claimed that it outperforms these traditional pair-wised ranking schemes. Similar list-wise method called RankCosine was proposed by Qin et al.[24].

Recently, Yue et al.[25] employed MAP as the optimal objective for the ranking problem. Due to MAP's special property that the whole document list must be considered not document pairs or single document, MAP objective can be classified as a list-wised approach. Yue's algorithm is called SVMMAP, and it was claimed that it had a good performance in ranking problem and a fast learning speed. In our work we will employ his SVMMAP in document summarization.

3. Motivation

Supervised extractive summarization can be recognized as ranking and selecting of sentences. The training data are some documents tagged with each sentence whether it is a summary or not, and the testing phrase is to tag new sentences using the model learned from training data.

As we have declared in Section 2, using LTK method in document summarization has the advantage of dealing with intrinsically different types of documents (Fig. 1 gives a simple illustration of the difference between traditional summarization model which consider all sentences with respect of the classification information and the LTK model which let sentences within a document to 'compete' with each other). Yet the ability of various LTK methods in generic document summarization is so far under-investigated extensively. In this section we will describe

some of the popular LTK schemes, both list-wised and pair-wised, for generic summarization and the features we will use.

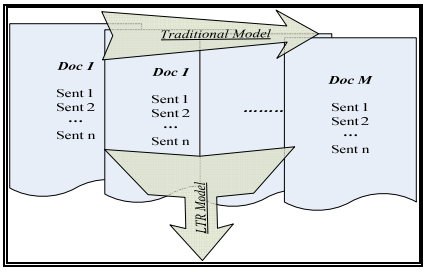

Figure 1:illustration of Learning to rank(LTR) summarization model

3.1. *Algorithms*

1): SVM

We choose SVM[26] as our baseline due to its sound theoretical fundations and wide usage. Other algorithms, both pair-wised and list-wised methods are also based on SVM.

The basic idea of SVM is to find a hyperplane which separates two classes of training data with the largest margine. SVM can be formalized as following:

Given n points training data D:

$$D = \{(x_i, y_i) \mid x_i \in R^p, y_i \in \{-1,1\}\}_{i=1}^{n} \quad (1)$$

where x_i stands for the p dimentional real value feature, y_i is the class(It is are a 2 classes problem). Then the hyperplane can be writen as :

$$\mathbf{w} \cdot \mathbf{x} - b = 0 \quad (2)$$

and by choosing weight vector **W** and the parameter b/||**w**||which determins the offset of the hyperplane from the origin along the weight vector, we get the maximum margine. More specifically, the opotimal hyperplane can be translated into the following problem:

$$\begin{aligned} &to\ \ Minimize: 1/2 \times \|\mathbf{w}\| + C\sum_{i=1}^{n} \xi_i \ (for\ \mathbf{w}, \xi \geq 0) \\ &subject\ to: y_i(\mathbf{w} \cdot x_i) \geq 1 - \xi_i \ (for\ \forall i \in n) \end{aligned} \quad (3)$$

Where the ξ_i is the slack variables, it will greater then 1 if the hypereplane gives wrong answer. The factor C controls the amount of regulairztion. SVM is a generalized linear classiffier, a special propertie of SVM is that it simultaniously minimazes the emperical classification error and maximizes the geometric margin.

2): Ranking SVMs

Ranking SVMs can be viewed as a generalized SVM that learns model not from binary labeled training data but from pair-wised preferences. It is claimed that for ranking problems, ranking SVMs takes the structure into consideration implicitly which the ordinary SVM is inappropriate to do.

Ranking SVMs can be formally defined as (we omit some parameters for simplicity):

$$\begin{aligned} &to\ \ Minimize: 1/2 \times \|\mathbf{w}\| + C\sum_{i,j} \xi_{i,j} \ (for\ \mathbf{w}, \xi \geq 0) \\ &subject\ to: (for\ \forall i, j \in n)\ \ (\mathbf{w} \cdot x_i - \mathbf{w} \cdot x_j) \geq 1 - \xi_{i,j} \end{aligned} \quad (4)$$

The n here stands for the size of the set of pair-wised preferences used for training. The weight vector w is the model we are going to learn from training data. When it is learned, the score of new sentence can be computed by $\mathbf{w} \cdot x$, which will used as ranking score. Ranking SVM is now state-of-the-art algorithm of sentence selection task, and claimed to be significantly outperforms regular SVM.

3): SVMMap

Mean Average Precision (MAP) is a widely used evaluation measure in IR. There were few learning algorithms optimizing directly for MAP, but neither could they find a global solution nor computational effective. Yue [9] gave an efficient algorithm SVMMAP, which is list-wised and based on SVM, to directly optimize according to MAP.

MAP can be formalized as following:

$$MAP(r1, r2) = \frac{1}{R} \sum_{i:r1_i=1} \Pr ec@i \quad (5)$$

where *r*1 and *r*2 mean two different kinds of ranking of a given documents(or other units). In equation (5) *r*1 is the real ranking, *r*2 is a system output to be evaluated. *R* is the total number of the relevant documents, which tagged as 1. Prec@i is the percentage of relevant documents which are correctly tagged in *r*2 at position i. Unlike ROCArea which gives equal penalties to each disordered relevant and non-relevant pairs, MAP assigns penalties to disordered units the higher the position the greater.

The difficulties of optimizing MAP loss function is that it is not decompose nicely into a sum of scores computed independently on each relative ordering of a positive/negative unit pairs. By using an interesting property of MAP that the loss function which is invariant to swapping two units with equal tag, [9] proposed an efficient algorithm to do the task. This also gave the algorithm the list-wise property which not possessed by other algorithms which define loss function according to F1 or ROCArea et al.

3.2. *Features*

Features are very important in summarization, Shen[14] has made a thorough investigation of the performances of CRF, HMM, and SVM. So, in order to simplify our work and make it comparable to the previous work, we shape our designation of features mainly under their framework.

We listed all the features used in Table 1. All these features were used in the work of [14].

TABLE I. THE FEATURES

Features	
Thematic	Position
Indicator	Length
Upper Case	Log Likelihood
Similarity to Neighboring Sentences	

4. Experiments

4.1. *Corpus & Evaluations*

To evaluate our approach, we applied the widely used test corpus of (DUC2001), which is sponsored by ARDA and run by NIST "http://www.nist.gov". The corpus DUC2001 we used contains 147 news texts, each of which has been labeled manually whether a sentence belongs to a summary or not. We design our experiment after the regular IR document retrieval approaches which tag document as relevant or not. The only preprocessing we did is to remove some stop words according to a stop word list.

We use F1 and ROUGE-2 score as the evaluation criteria. 10-fold cross validation is used in order to reduce the uncertainty of the model we trained. The final F1&ROUGE-2 score reported is the average of all these 10 experiments.

4.2. *Results*

TABLE II. EXPERIMENT RESULTS

	CRF	***SVM***	***SVMRank***	***SVMMAP***
ROUGE-2	0.455	0.417	0.434	**0.460**
F1	0.389	0.343	0.372	**0.394**

The experiment results are given at Tab. II. We can clearly see that our approach gained good results both in ROUGE-2 and F1 scores. Compared with regular SVM, the pair-wised approach SVMRank has about 8.5% improvement in F1 and 4% in ROUGE-2. The results of our approach, by using the list-wised SVMMAP, is about 14% and 10% higher than SVM in F1 and ROUGE-2, 6% higher than SVMRank both in F1 and ROUGE-2. The best reported results using the same features, the CRF approach, which was conducted by Shen[14], is also a lightly weaker than ours.

TABLE III. COMPARED WITH UNSUPERVISED APPROACHES

	Random	***LEAD***	***LSA***	***HITS***	***SVMMAP***
ROUGE-2	0.245	0.377	0.382	0.431	**0.460**
F1	0.202	0.311	0.324	0.368	**0.394**

We also compared our results with some well known unsupervised approaches including Random, LEAD, LSA and HITS. Our approach also well outperformed these approaches. The results were shown in Tab III.

5. Conclution and Future Work

In this paper we described an interesting extractive generic summarization approach which based on a list-wised LTK scheme SVMMAP. The experiment results show that SVMMAP can do well in document summarization. We compared our result with the state-of-the-art pair-wised LTK approach SVMRank, and our result greatly outperform it.

Although our work is promising, it is still very primitive, in our future work, we will investigate more LTK approaches, and get study the effect of different features in our summarization system.

References

[1] Luhn, H., *The automatic creation of literature abstracts,* . IBM J Res. Develop, 1959. **2**(2): p. 159-165.

[2] Radev, D., E. Hovy, and K. McKeown, *Introduction to the special issue on summarization.* Computational linguistics, 2002. **28**(4): p. 399-408.

[3] Zong, C., ed. *Statistical Natural Language Processing*. 2008, Tsinghua University Press.

[4] Yeh, J., et al., *Text summarization using a trainable summarizer and latent semantic analysis.* Information Processing and Management, 2005. **41**(1): p. 75-95.

[5] Daume III, H. and D. Marcu. *Bayesian query-focused summarization*. 2006: Association for Computational Linguistics.

[6] Wu, X. and C. Zong. *A New Approach to Automatic Document Summarization*. in *International Joint Conference on Natural Language Processing (IJCNLP)*. 2008. Hyderabad, India.

[7] Liu, T.-Y., ed. *Learning to Rank for Information Retrieval*. Foundations and Trends in Information Retrieval. Vol. 3. 2009. 225-331.

[8] Joachims, T. *Optimizing Search Engines using Clickthrough Data*. in *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining*. 2003. New York.

[9] Yue, Y., et al. *A Support Vector Method for Optimizing Average Precision*. in *SIGIR*. 2007.

[10] Kupiec, J., J. Pedersen, and F. Chen, *A trainable document summarizer*. Research and Development in Information Retrieval, 1995: p. 68-73.

[11] Lin, C. *Training a selection function for extraction.* 1999: ACM New York, NY, USA.

[12] Jing, H. and K. McKeown. *The decomposition of human-written summary sentences*. in *Proceedings of the 22nd Conference on Research and Development in Information Retrieval (SIGIR-99),*. 1999. Berkeley, CA: ACM New York, NY, USA.

[13] Osborne, M. *Using maximum entropy for sentence extraction*. 2002: Association for Computational Linguistics.

[14] Shen, D., et al. *Document summarization using conditional random fields*. 2007.

[15] Wang, C., et al. *Learning query-biased web page summarization*. in *Conf. on Information and Knowledge Management*. 2007.
[16] Metzler, D. and T. Kanungo. *Machine Learned Sentence Selection Strategies for Query-Biased Summarization*. in *SIGIR Learning to Rank Workshop*. 2008.
[17] Li, P., C.J. Burges, and Q.W. Mcrank. *Learning to rank using multiple classification and gradient boosting*. in *In Proc. 21st Proc. of Advances in Neural Information Processing Systems*. 2007.
[18] Crammer, K. and Y. Singer. *Pranking with ranking*. in *NIPS*. 2002.
[19] Cao, Z., et al. *Learning to rank: From pairwise approach to listwise approach*. in *ICML*. 2007. Corvallis, OR.
[20] Xia, F., et al. *Listwise approach to learning to rank - theory and algorithm*. in *In ICML '08: Proceedings of the 25th international conference on Machine learning,*. 2008. New York, NY.
[21] Burges, C.J.C., et al. *Learning to Rank using Gradient Descent*. in *ICML*. 2005. Bonn,Germany.
[22] Freund, Y., et al. *An efficient boosting algorithm for combining preferences*. in *In Machine Learning: Proceedings of the Fifteenth International Conference*. 1998.
[23] Herbrich, R., T. Graepel, and K. Obermayer, eds. *Large margin rank boundaries for ordinal regression* In Advances in Large Margin Classifiers. 2000.
[24] Qin, T., et al., *Query-level loss functions for information retrieva.* Information Processing & Management, 2007.
[25] Yue, Y., et al. *A support vector method for optimizing average precision.* in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. 2007. New York, NY.
[26] Vapnik, V., *The nature of statistical learning theory*. 2000: Springer Verlag.