ISTANBUL UNIVERSITY –
JOURNAL OF ELECTRICAL & ELECTRONICS ENGINEERING
YEAR : 2004 (1101-1109)
VOLUME : 4
NUMBER : 2

ARG: A TOOL FOR AUTOMATIC REPORT GENERATION

K. Murat KARAKAYA [1] **H. Altay GÜVENİR** [2]

[1,2] Bilkent University, Department of Computer Engineering
06533 Bilkent, Ankara, TURKEY
Tel: +90 (312) 266 41 33, Fax: +90 (312) 266 41 26

[1]E-mail: muratk@cs.bilkent.edu.tr [2]E-mail: guvenir@cs.bilkent.edu.tr

ABSTRACT

The expansion of on-line text with the rapid growth of the Internet imposes utilizing Data Mining techniques to reveal the information embedded in these documents. Therefore text classification and text summarization are two of the most important application areas. In this work, we attempt to integrate these two techniques to help the user to compile and extract the information that is needed. Basically, we propose a two-phase algorithm in which the paragraphs in the documents are first classified according to given topics and then each topic is summarized to constitute the automatically generated report.

Keywords: : *Data mining, text summarization, text classification, automatic report generation.*

1. INTRODUCTION

The explosive growth of the Internet has made millions of on-line documents easily accessible to almost every user. However, it has become almost impossible to take the full advantage of information embedded inside these documents without the help of various data mining techniques and tools. In this work, we propose a novel approach to reveal the information stored in many documents by integrating two important data mining techniques: text classification and text summarization.

In our observation, these two techniques could be more powerful and beneficial if they are integrated and employed together. User information requirement can not be simplified only by supporting text categorization and text summarization methods, because user wants not only to classify text or to summarize a single document but to locate the main ideas and important facts contained in numerous documents easily and completely. Therefore, we try to put a step forward from these two basic methods in order to satisfy user expected information need by integrating these methods.

2. MOTIVATION

As an example, assume that we need to find out the latest news about economic crisis in Turkey. In order to accomplish this, we can explore a news web site using a search engine with some suitable keywords. Probably, it can return us many articles ordered according to their relevance. Nevertheless, most of the time, we do not want to read all the articles because of various reasons; e.g., we do not have enough time, only small number of them is directly related to the subject, or many of the retrieved documents have repeated information. Moreover, for sometimes, we may not know what we are looking for specifically. Continuing with the example, we can read several of retrieved news articles and get an overall idea of the subject.

Received Date : 09.07.2001
Accepted Date: 15.06.2004

Then, we can define our information need more specifically such as "the causes and results of the crisis, and the actions taken so far". We expect to get important and crucial points in the news about these topics. In reality, we do not want to read all news articles. At this point, we propose to use a tool, which can handle such a situation and help the user. This tool will anticipate your specific information needs and after examining documents it can supply you related and important pieces of information embedded in different documents under the titles of the topics. We name this process Automatic Report Generation (ARG).

3. BACKGROUND AND RELATED WORK

As the dramatic expansion of on-line text available on the Internet, the development of methods for automatically categorizing text becomes more important. Text categorization may be defined as the problem of assigning free text documents to predefined categories. A growing number of statistical learning methods have been applied to this problem in recent years, including regression models[7,27], nearest neighbor classifiers[11,14,25,26,28], Bayes belief networks[3,9,10,12,16,17,18,23], decision trees[2,9,12,18], neural networks[22,24] and inductive rule learning techniques[1,5,6,19]. Recent studies have proved the success of statistical approaches for learning to classify text documents [3,10,16,20,26]. These approaches commonly represent documents as vectors of words, and learn by gathering statistical information from the observed frequencies of these words within documents belonging to the different clusters.

Most work in text classification have involved articles taken from a news resource, or a medical database. In these instances, human experts choose correct topic labels. The domain of USENET newsgroup posting is a common test bed for text classification. The labels here are just the newsgroups to which the documents were originally posted. The web directories, such as AltaVista or Yahoo!, are other common text categorization tests resources.

A summary is simply defined as an abbreviated version of a document. In other words, a summary only contains the crucial information elements needed to comprehend the given text. This is however not a very sensible definition for many cases, because a summary requires to satisfy the information the user needs, and this information will not be the same for every user. Therefore we can assert that there is an information gap between the summary and user real need.

In automatic text summarization a computer program produces automatically an abstract or summary from an original source text. There are two distinct techniques; text extraction or text abstraction[13]. Text extraction means to extract pieces of an original text on a statical basis or with heuristic methods and put together it to a new shorter text with the same information content.

In text extraction the method is basically to give scores to each sentence depending on the importance of each sentence. When creating the summary the most significant sentences are kept. The scores can be based on high-frequent words, bold or numerical text, proper nouns, citations, position in text etc.

On the other hand, text abstraction is the process of parsing the original text in a linguistic way, interpreting the text and finding new concepts to describe the text and then generating a new shorter text with the same information content. In this work we use a text extraction method to summarize the topics. The implementation of the extraction method is provided in implementation section.

In [21], a method similar to our work is proposed. The authors also suggest combining two data mining techniques, text classification and text summarization. They propose a text clustering algorithm to automatically classify the documents and a text summarization algorithm to summarize a document in the clusters. After clustering the given documents, they supply the important indexes for each class to give an idea to the user about the documents in the cluster. The user can select one of the documents in the cluster and a summary of this document is produced automatically.

The contributions of the work are as follows. First of all, we consider the paragraphs, not the whole document, as the granularity of information for both in text classification and text summarization. The choice of selecting paragraphs is based on the idea that paragraphs have more homogenous and unified ideas than the documents do. In a document there can be several different ideas of a larger topic where as a paragraph presents information of a more specific point of view. In the past work, to the best of our knowledge, the paragraph structure is not used in both data mining techniques. Secondly, unlike previous work, we do not try to summarize a single document but we aim to summarize a collection of paragraphs of different documents in a cluster. Thus, the user can get a complete summary of the clustered documents, so that he can grasp a better knowledge embedded in different documents. Third, we use a supervised text classification method in which the user supplies several topics with sample paragraphs. This method enables the user to express his information need better than using keywords. As a result our method can distinguish the important information pieces in different documents and output them according to subject topics determined by user thereby removing the information gap between summary and user.

4. TEMPLATE ALGORITHM

We can formalize the process of automatic report generation as follows (Fig. 1). Assume that there is a collection of text documents related to a common subject. We take for granted that user has read some of these documents and determined the information needs as subject topics. User supplies related paragraphs in the documents (s)he read as representative information to each subject topics. These paragraphs are used in text classification as the training set. Other documents are split into paragraphs and then they are categorized into predefine subject topics. At the last step of the algorithm, each subject topic is summarized and combination of these summaries constructs the report that is generated automatically.

We can implement many variations of the template algorithm using different categorization and summarization techniques available in data mining. In the following section we explain one simple but demonstrative version of the template algorithm.

1. User determines subject topics (classes)
2. User splits one or more article into their paragraphs and distributes them into topics
3. Each topic (class) is indexed using paragraphs in step 2
4. Other documents are split into paragraphs
5. Paragraphs (in step 4) are classified according to given classes (in step 1)
6. Each topic is summarized
7. Summaries are compiled and outputted as a report

Figure 1. Template Algorithm for ARG

5. IMPLEMENTATION DETAILS

In this section, the implementation details of proposed algorithm is given. In the first step the user defines the information need by supplying topic names associated with several paragraphs. Then, the paragraphs of the other documents are classified. In the last step, the summarizations of each topic are compiled and automatically generated report is produced. We use Perl as the programming language.

5.a. Determining Report Main Topics

We enable the user to split an article into its paragraphs. The user splits one or more articles and then groups their paragraphs according the subject topics, which is defined by himself/herself. This is the only information given by the user to the system to produce the report. In the other steps of the method, the processes are executed automatically.

5.b. Classification of Paragraphs

In order to implement classification phase of the ARG Tool, we have used Rainbow [15], which is a program that performs statistical text classification using naive Bayes algorithm. To prepare the compiled articles we first split them into paragraphs and enumerate them according to their article number. For example, if the first article is stored in file 1.txt, its third paragraph is stored in the file d1p3.txt. Then, we input the subject topics and the related paragraphs provided by the user as the classes and training set to Rainbow. Rainbow indexes these classes according to given paragraphs. That is, Rainbow reads the paragraphs in each topic and archives a "model" containing their statistics. Once indexing is performed and a model has been archived to disk, Rainbow can perform document classification. Statistics from a set of training paragraphs will determine the parameters of the classifier; classification of a set of testing documents will be output. We use other documents' split paragraphs as the test case and record the classification result.

5.c. Summarizing And Compilation

After evaluating all the paragraphs in the documents and distributing them to related topics, the next step of the Automatic Report Generation takes place. In that phase, to summarize each topic we determine which paragraph(s) has the highest importance to their topics by using a simple score function. We use an automatic text summarization method based on text extraction. After Rainbow creates a model for given topics using training paragraphs, it can return us the words and their probabilities for each topic. The more frequently a word is used in that topic, the higher the probability is. For ranking each paragraph in test set after classification, we count its words and sum up its corresponding probabilities as its score. Then, we normalize the score of each paragraph by dividing the total score of the paragraph by the number of its words. The higher the score is the higher the rank of the paragraph. A similar method is used in Microsoft word processor as AutoSummarize feature [8]. After finding scores for each paragraph, we can generate the report by selecting predetermined numbers of paragraphs from each topic according to their ranking and integrating them under the topic names.

6. EXPERIMENTS AND RESULTS

As an example, assume that we are trying to learn what have happened in Turkey after reading news about economic crisis. For this purpose, we execute a search at CNN Europe web site (CNN, 2001) using the search key words "turkey economic crisis". But the search engine returns 250 (max. number) news articles about the subject. We download only eleven of them, which seem to be most related according to their titles1 (Table 1). Then we simply only read two of them (1.txt and 8.txt).

Table 1: Titles of selected news articles

ARTICLE NO.	TITLE
1.txt:	Turkey moves to stabilize currency
2.txt:	Turkey promises new economic measures
3.txt	Ecevit rides out political storm
4.txt	EU welcomes Turkey reforms
5.txt	Turkey braced for falling currency
6.txt	Turkey economy plan unveiled
7.txt	Turkey hopeful of financial recovery
8.txt	Turkey's currency tumbles in crisis
9.txt	Turkey to restructure state banks to rescue economy
10.txt	Turkey seeks to soothe markets
11.txt	Turkey to unveil urgent economic action, no program yet

After reading these two articles we determine our information under three topics: causes of crisis, actions taken and the results seen. Then we split article 1 and 8 into paragraphs and assign each of them to the subject topics as in Table 2.

Table 2: Partitioning of paragraphs into subject classes. Note that p15.txt file holds the fifth paragraph of the first article.

Causes	Results	Measures taken
d1p5.txt	d1p3.txt	d1p1.txt
d8p3.txt	d1p4.txt	d1p2.txt
d8p4.txt	d8p1.txt	d8p5.txt
	d8p2.txt	
	d8p6.txt	
	d8p7.txt	

After constructing classes and determining training paragraphs sets, we run Rainbow to index the data and to build the model. Then each remaining article has been split and 50 paragraphs are produced totally. We run Rainbow to classify 50 paragraphs. We record the results and the first phase of the algorithm is finished. The ten indexing terms of each topic with the highest probabilities are given at Table 3.

[1] We limit the number of documents so small that we can compare the content of the documents and the report generated.

Table 3: Indexes for the subjects

Causes		**Results**		Measures taken	
Government	0,017	Imf	0,0136	percent	0,0123
Ecevit	0,013	Lira	0,0136	ecevit	0,0082
Sezer	0,011	percent	0,0106	government	0,0082
Corruption	0,009	government	0,0091	economic	0,0082
Country	0,009	turkish	0,0091	minister	0,0082
President	0,009	turkey	0,0091	meeting	0,0082
Failing	0,007	programme	0,0091	officials	0,0082
Political	0,007	Rate	0,0091	turkish	0,0062
Turkey	0,007	dollar	0,0091	inflation	0,0062
Crisis	0,007	central	0,0076	turkey	0,0062
TOTAL NUMBER OF INDEXES: 141		**TOTAL NUMBER OF INDEXES: 206**		**TOTAL NUMBER OF INDEXES: 98**	

Table 4: Rankings of paragraphs

Causes			**Results**			**Measures taken**		
Rank	**Prg.No.**	**Score**	**Rank**	**Prg.No.**	**Score**	**Rank**	**Prg.No.**	**Score**
1	d5p3.txt	0,00183	1	d5p1.txt	0,00174	1	d5p4.txt	0,00141
2	d10p3.txt	0,00110	2	d9p3.txt	0,00134	2	d5p2.txt	0,00133
3	d10p1.txt	0,00085	3	d5p6.txt	0,00128	3	d2p1.txt	0,00077
4	d7p5.txt	0,00084	4	d5p5.txt	0,00125	4	d10p4.txt	0,00075
5	d2p3.txt	0,00062	5	d7p4.txt	0,00107	5	d2p5.txt	0,00069
6	d3p3.txt	0,00059	6	d4p5.txt	0,00098	6	d3p6.txt	0,00056
7	d10p2.txt	0,00057	7	d6p1.txt	0,00088	7	d9p6.txt	0,00049
8	d9p5.txt	0,00049	8	d11p2.txt	0,00073	8	d6p4.txt	0,00047
9	d10p5.txt	0,00040	9	d2p4.txt	0,00073	9	d7p1.txt	0,00046
10	d3p2.txt	0,00039	10	d7p3.txt	0,00072	10	d11p4.txt	0,00042
11	d7p2.txt	0,00014	11	d4p1.txt	0,00067	11	d2p2.txt	0,00042
			12	d3p4.txt	0,00065	12	d7p6.txt	0,00041
			13	d11p5.txt	0,00064	13	d3p5.txt	0,00033
			14	d11p6.txt	0,00058	14	d9p2.txt	0,00009
			15	d11p1.txt	0,00054			
			16	d3p1.txt	0,00052			
			17	d6p3.txt	0,00051			
			18	d9p1.txt	0,00047			
			19	d9p4.txt	0,00045			
			20	d9p7.txt	0,00041			
			21	d11p3.txt	0,00034			
			22	d6p2.txt	0,00029			
			23	d4p4.txt	0,00024			
			24	d4p2.txt	0,00022			
			25	d4p3.txt	0,00014			

In automatic summarization phase, we utilize Rainbow indexing information. The word-vectors for each classification are used to scoring paragraphs. Simply we compute each paragraph's words probabilities and divide the total score by the word length of the paragraph. The obtained normalized scores are presented in Table 4.

After getting the ranking of paragraphs we can summary every topic by selecting top n paragraphs. User can specify the number of paragraphs in the summary or give a percentage of the total documents' size. For the experiment we assume that two paragraphs are enough to represent the information in each class.

Unfortunately there are no objective criteria for evaluating the produced report. However, two main phases can be evaluated independently. The success of categorization phase mostly depends on the given paragraphs. If these paragraphs represent the required information well enough then we can extract important indexing words. Also that information affects the summarization step though we use the indexing terms in scoring the paragraphs to be selected. However, this relation is not assumed to be only negative, it can produce positive effects. If user selects good sample data, which represents his/her need, then in the end probably (s)he gets what (s)he expected. In classical summary techniques, user is out of the process, waits for the resulting text. Above dependency between sample paragraphs and automatic report generation is an advantage if it is used properly.

For the time being we evaluate the resulting report by producing another report (See Appendices A, B, and C). The second report consists of the paragraphs with the lowest ranking. It is seen that generated report satisfies user information need and is much more related to the topics than the second report. To give an idea for the efficiency of the whole algorithm, the running time of two phases is recorded. We run our implementation and Rainbow on a Sparc Station 20 with 60 MHz CPU and 128 MB RAM. Splitting documents into paragraph takes 0.08 seconds on the average. We remind that there are 11 documents with total size 26KB. The classification process takes 2.69 seconds whereas summarization process takes 0.27 seconds on the average.

7. Limitations

In classification of paragraphs, the size and number of training set are supposed to be small because the user is not reluctant to read more than a few documents. That would yield not so good vocabulary indexing and classification results. The human interpretation in the initial classification of representative paragraphs might not create a good basis for paragraph classification. Furthermore, sometimes-even user could be not sure what he/she really expects as a resulting report. In these cases, we think that unsupervised classification will fit the situation better. After automatic classification of paragraphs, we can summarize these classes into new paragraphs and return the user by using their most important keywords as their topics. Then, we expect user to select and compile these summarized paragraphs to build his/her own report.

The summarization step of the template algorithm is implemented simply. We exploit the paragraphs to summarize the each subject class content. So, user will have a report constituted of selected paragraphs. As it is discussed in Future Work, there could be different alternatives for summarization to be implemented.

8. Conclusion And Future Work

In this work, we aim to propose a framework and implement a simple version of it, which enables a user to compile an automatically generated report on specific subjects. We suggest using a two-phase algorithm. In order to implementing the proposed method, we make use of the two important techniques used in Data Mining: text categorization and text summarization. We implement the proposed template algorithm and test it for several different document compilations. The results are satisfactory if the simple implementation and easy use of the system are taken into consideration.

For the future work, we plan to improve the interface. For the time being, user interacts with the system through Unix command shell and Perl scripts. Future implementation will have windows driven interface. Another thing should be done as a future work, is to implement different summarization techniques and let the user decide which one to use. For example, user can want to see only segments of important statements. Furthermore, to minimize the user intervention, the classification of paragraphs can be done without user help. For this reason automatic text clustering methods can be used. For evaluation of the method, we plan to add more numerical analysis methods. For example, an expert can rank the categorized paragraphs and then the results can be compared accordingly.

REFERENCES

[1] Apte,C., Damerau,F., and Weiss,S., Towards language independent automated learning of text categorization models, In Proceedings of the 17th Annual ACM/SIGIR conference, 1994.

[2] Apte,C., Damerau,F., and Weiss,S., Text mining with decision rules and decision trees, In Proceedings of the Conference on Automated Learning and Discorery, Workshop 6: Learning from Text and the Web, 1998.

[3] Baker,L.D., Mccallum,A.K., Distributional clustering of words for text categorization, In Proceedings of the 21th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'98), pages 96-103, 1998.

[4] CNN Europe Web site, http://europe.cnn.com/, 2001.

[5] Cohen,William W., Text categorization and relational learning, In The Twelfth International Conference on Machine Learning (ICML'95), Morgan Kaufmann, 1995.

[6] Cohen,William W., Singer,Yoram, Context-sensitive learning methods for text categorization, In SIGIR '96: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 307-315, 1996.

[7] Fuhr,N., Hartmanna,S., Lustig,G., Schwantner,M., and Tzeras,K., Air/x - a rule-based multistage indexing systems for large subject fields, In 606-623, editor, Proceedings of RIAO'91, 1991.

[8] Gore, K., Cogito Auto Sum: What less can we say? http://slate.msn.com/features/cogitoautosum/cogitoautosum.asp, 2001.

[9] Joachims,Thorsten, Text Categorization with Support Vector Machines: Learning with Many Relevant Features, In European Conference on Machine Learning (ECML), Berlin, pages 137-142, 1998. Springer.

[10] Koller,D. and Sahami,M., Hierarchically classifying documents using very few words, In The Fourteenth International Conference on Machine Learning (ICML'97), pages 170-178, 1997.

[11] Lam ,W. and Ho,C.Y., Using a generalized instance set for automatic text categorization, In Proceedings of the 21th Ann. Int. ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'98), pages 81-89, 1998.

[12] Lewis,D.D. and Ringuette,M., Comparison of two learning algorithms for text categorization, In Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval (SDAIR'94), 1994.

[13] Lie, D.H., Sumatra, A system for Automatic Summary Generation, Carp Technologies, Hengelosestraat 174, 7521 AK Enschede, 2001. http://www.carp-technologies.nl/SumatraTWLT14paper/SumatraTWLT14.html

[14] Masand,B., Linoff, G., and Waltz,D., Classifying news stories using memory based reasoning, In 15th Ann. Int. ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'92), pages 59-64, 1992.

[15] McCallum, Andrew Kachites, Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering, 1996. http://www.cs.cmu.edu/~mccallum/bow.

[16] McCallum A. and Nigam,K., A comparison of event models for naive bayes text classification, In AAAI-98 Workshop on Learning for Text Categorization, 1998.

[17] Mitchell, T.M., Machine Learning, The McGraw-Hill Companies, Inc., 1997.

[18] Moulinier,I., Is learning bias an issue on the text categorization problem? In Technical report, LAFORIA-LIP6, Universite Paris VI, 1997.

[19] Moulinier,I., Raskinis,G. and Ganascia,J., Text categorization: a symbolic approach, In Proceedings of the Fifth Annual Symposium on Document Analysis and Information Retrieval, 1996.

[20] Nigam, K., McCallum, A., Thrun, S. and Mitchell, T., Learning to Classify Text from Labeled and Unlabeled Documents, In Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI-98), pages 792-799, 1998.

[21] Neto, J. Larocca, Santos, A.D., Kaestner, C.A.A., Freitas, A.A., Document clustering and text summarization, Proc. 4th Int. Conf. Practical Applications of Knowledge Discovery and Data Mining (PADD-2000), sh=41-55, London: The Practical Application Company, 2000.

[22] Ng,H.T., Goh,W.B., and Low,K.L., Feature selection, perceptron learning, and a usability case study for text categorization,In 20th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'97), pages 67-73,1997.

[23] Tzeras,K. and Hartman,S., Automatic indexing based on bayesian inference networks, In Proc 16th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'93), pages 22-34, 1993.

[24] Wiener,E., Pedersen,J.O., and Weigend,A.S., A neural network approach to topic spotting, In Proceedings of the Fourth Annual Symposium on Document Analysis and Information Retrieval (SDAIR'95), pages 317-332, Nevada, Las Vegas, 1995. University of Nevada, Las Vegas.

[25] Yang ,Y., Expert network: Effective and efficient learning from human decisions in text categorization and retrieval, In 17th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'94), pages 13-22, 1994.

[26] Yang,Y., An evaluation of statistical approaches to text categorization. Journal of Information Retrieval, 1(1/2):67-88, 1999.

[27] Yang,Y. and Chute,C.G., An example-based mapping method for text categorization and retrieval, ACM Transaction on Information Systems (TOIS), 12(3):252-277, 1994.

[28] Yang,Y. and Pedersen,J.P., A comparative study on feature selection in text categorization, In Jr. D. H. Fisher, editor, The Fourteenth International Conference on Machine Learning, pages 412-420. Morgan Kaufmann, 1997.

APPENDIX A: HIGHEST AND LOWEST RANK P ARAGRAPH FOR CAUSES TOPIC

HIGHEST: The financial crisis was sparked by a bitter dispute between Prime Minister Bulent Ecevit and President Ahmet Necdet Sezer, which escalated earlier this week. Ecevit stormed out of a meeting with Sezer, upset after the president criticized the government for failing to battle corruption. Political stability is seen as essential to the success of the programme, and the clash between the country's two top leaders led to fears of a government collapse. Ecevit's three-party coalition government is Turkey's most stable in years, but the row terrified Turkish banks, many of them weakened from the country's financial crisis Iate last year. Corruption allegations panicked investors drained more than $7 billion from central bank reserves on Monday, leading the central bank to cut off the taps to them in a bid to wrest those dollars back, sending key lending rates over 4,000 percent.

LOWEST: The stock market was also buoyed by news that the two main characters whose argument triggered the latest financial downturn, the Prime Minister Bulent Ecevit and President Ahmet Necdet Sezer, were set to meet. The talks, the first between the two, follow a public bust-up 10 days ago over the pace of anti-corruption measures --an argument which Ecevit later described as a "serious crisis."

APPENDIX B: HIGHEST AND LOWEST RANK P ARAGRAPH FOR RESULT TOPIC

HIGHEST: Turkey is facing a day of turbulence after the government decided to abandon its exchange-rate control s --effectively allowing a devaluation of the lira. It announced it was floating its currency in a dramatic move to deal with one of its worst financial crises in decades. Analysts in Asia's main financial centers were predicting a sharp fall in the value of the Turkish lira of between 30 and 40 percent on the opening of markets on Thursday. The government abandoned its defense of the Turkish lira that had already cost it $3 billion and driven overnight interest rates to 4,000 percent.

LOWEST: The EU, while offering encouragement to Ankara, has yet to be convinced that Turkey is a truly democratic country ready to join the likes of France, Germany and Sweden. Turkey was accepted as an official candidate for the EU in 1999 but has yet to begin accession negotiations because of concerns over its human rights record.

APPENDIX C: HIGHEST AND LOWEST RANK P ARAGRAPH FOR MEASURES TOPIC

HIGHEST: The exchange-rate decision was issued by Turkish authorities in a statement released after a 10-hour meeting between Ecevit, his top ministers and Central Bank officials. "Due to the recent economic conditions, the exchange rate will be left to float freely," the statement read. Economy Minister Recep Onal said the decision was effective Thursday. The economic stabilization programme has been under pressure since November when international investors were scared off by allegations of corruption in the banking sector.

LOWEST: Banking reform is a key area of the new program now under discussion by Turkish ministers and officials. Economy Minister Kemal Dervis promised Tuesday that the new program would be ready soon, but declined to give a date. "W e are still discussing various proposals and we want to announce the program as a whole when it is finished. I ask you to be patient," he said.