Automatic Text Summarization with Neural Networks

Khosrow Kaikhah

Abstract—**A novel technique for summarizing news articles using neural networks is presented. A neural network is trained to learn the relevant features of sentences that should be included in the summary of the article. The neural network is then modified to generalize and combine the relevant features apparent in summary sentences. Finally, the modified neural network is used as a filter to summarize news articles.**

Index Terms—**adaptive clustering, feature fusion, neural networks, pruning, text summarization**

I. Introduction

Text summarization has become an important and timely tool for assisting and interpreting text information in today's fast-growing information age. There is an abundance of text material available on the Internet, however, usually the Internet provides more information than is needed. Therefore, a twofold problem is encountered: searching for relevant documents through an overwhelming number of documents available, and absorbing a large quantity of relevant information. Summarization is a useful tool for selecting relevant texts, and for extracting the key points of each text. Some articles such as academic papers have accompanying abstracts, which present their key points. However, news articles have no such accompanying summaries, and their titles are often not sufficient to convey their important points. Therefore, a summarization tool for news articles would be extremely useful, since for a given news topic or event, there are a large number of available articles from the various news agencies and newspapers. Because news articles have a highly structured document form, important ideas can be obtained from the text simply by selecting sentences based on their attributes and locations in the article.

We propose a machine learning approach that uses artificial neural networks to produce summaries of arbitrary length of news articles. A neural network is trained on a corpus of articles. The neural network is then modified, through feature fusion, to produce a summary of highly ranked sentences the article. Through feature fusion, the network discovers the importance (and unimportance) of various features used to determine the summary-worthiness of each sentence. The input to the neural network can be either real or binary vectors.

II. Literature Review

There are two divergent approaches to automatic text summarization: 1) summarization based on abstraction, wherein the text has to be understood, and the summary produced from such an understanding, and 2) summarization based on extraction, which involves selecting a number of important sentences from the source text. Summarization by abstraction is concerned with issues related to text understanding, semantic representation and modification, and natural language processing. A review of the abstraction approach can be found in [1]. Extraction is mainly concerned with judging the importance, or indicative power, of each sentence in a given document. Typically, sentences are scored in terms of their importance in the document. A summary can then be obtained by choosing a number of top scoring sentences. Our approach is based on extraction by sentence ranking.

The first step in summarization by extraction is the identification of important features. There are two distinct types of features: non-structured features (paragraph location, offset in paragraph, number of bonus words, number of title words, etc.) and structured features (rhetorical relations between units such as cause, antithesis, condition, contrast, etc.) [2]. One group of researchers utilize only non-structured features; such features include sentence length [3], sentence location [4] [5], term prominence [6] [7], presence of cue words or phrases [8] [9], presence of words occurring in title [8], and presence of proper names [3]. On the other hand, a group of researchers attempt to exploit structural relations between units of consideration. Marcu in [10] built a Rhetorical Structure Tree to represent rhetorical relations between sentence segments of the document; Mani in [1] used a similar approach, but considered only localized relations due to the high cost of considering global relations. Since we do not expect a dense enough network of rhetorical or structural relations between sentences in news articles, we use only non-structured features and identify a set of features that can be used in a ranking system. In our approach, we utilize a feature fusion technique to discover which features out of the available ones are actually useful, without manual intervention.

III. Features

Each document is converted into a list of sentences. Each sentence is represented as a vector $[f_1,f_2,...,f_7]$, composed of 7 features.

Khosrow Kaikhah is with the Computer Science Department, Texas State University, San Marcos, Texas 78666 USA (email: kk02@TxState.edu)

0-7803-8278-1/04/$20.00 ©2004 IEEE

TABLE I
SEVEN FEATURES OF A DOCUMENT

Feature	Description
f_1	Paragraph follows title
f_2	Paragraph location in document
f_3	Sentence location in paragraph
f_4	First sentence in paragraph
f_5	Sentence length
f_6	Number of thematic words in the sentence
f_7	Number of title words in the sentence

Features f_1 to f_4 represent the location of the sentence within the document, or within its paragraph. It is expected that in structured documents such as news articles, these features would contribute to selecting summary sentences. Brandow et al. in [11] have shown that summaries consisting of leading sentences outperform most other methods in this domain, and Baxendale in [4] demonstrated that sentences located at the beginning and end of paragraphs are likely to be good summary sentences. Feature f_5, sentence length, is useful for filtering out short sentences such as datelines and author names commonly found in news articles. We also anticipate that short sentences are unlikely to be included in summaries [3]. Feature f_6, the number of thematic words, indicates the number of thematic words in the sentence, relative to the maximum possible. It is obtained as follows: from each document, we remove all prepositions, and reduce the remaining words to their morphological roots [12]. The resultant *content words* in the document are counted for occurrence. The top 10 most frequent content words are considered as thematic words. This feature determines the ratio of thematic words to content words in a sentence. This feature is expected to be important because terms that occur frequently in a document are probably related to its topic [6]. Therefore, we expect a high occurrence of thematic words in salient sentences. Finally, feature f_7 indicates the number of title words in the sentence, relative to the maximum possible. It is obtained by counting the number of matches between the content words in a sentence, and the words in the title. This value is then normalized by the maximum number of matches. This feature is expected to be important because the salience of a sentence may be affected by the number of words in the sentence also appearing in the title. These features may be changed or new features may be added. The selection of features plays an important role in determining the type of sentences that will be selected as part of the summary and, therefore, would influence the performance of the neural network.

IV. Text Summarization Process

There are three phases in our process: neural network training, feature fusion, and sentence selection. The first step involves training a neural network to recognize the type of sentences that should be included in the summary. The second step, feature fusion, prunes the neural network and collapses the hidden layer unit activations into discrete values with identified frequencies. This step generalizes the important features that must exist in the summary sentences by fusing the features and finding trends in the summary sentences. The third step, sentence selection, uses the modified neural network to filter the text and to select only the highly ranked sentences. This step controls the selection of the summary sentences in terms of their importance. These three steps are explained in detail in the next three sections.

A. Phase I: Neural Network Training

The first phase of the process involves training the neural networks to learn the types of sentences that should be included in the summary. This is accomplished by training the network with sentences in several test paragraphs where each sentence is identified as to whether it should be included in the summary or not. This is done by a human reader. The neural network learns the patterns inherent in sentences that should be included in the summary and those that should not be included. We use a three-layered feedforward neural network, which has been proven to be a universal function approximator. It can discover the patterns and approximate the inherent function of any data to an accuracy of 100%, as long as there are no contradictions in the data set.

We use a gradient method for training the network where the energy function is a combination of error function and a penalty function. The goal of training is to search for the global minima of the energy function. The total energy function to be minimized during the training process is:

$$\theta(w,v) = E(w,v) + P(w,v) \qquad (1)$$

The error function to be minimized is the mean squared error.

$$E(w,v) = \frac{1}{L}\sum_{l=i}^{L}\sum_{k=1}^{n}(o_{lk} - d_{lk})^2 \qquad (2)$$

The addition of the penalty function drives the associated weights of unnecessary connections to very small values while strengthening the rest of the connections. Therefore, the unnecessary connections and neurons can be pruned without affecting the performance of the network. The penalty function is defined as:

$$P(w,v) = \rho_{decay}\left(P_1(w,v) + P_2(w,v)\right) \qquad (3)$$

$$P_1(w,v) = \varepsilon_1\left(\sum_{j=1}^{h}\sum_{i=1}^{m}\frac{\beta w_{ij}^2}{1+\beta w_{ij}^2} + \sum_{j=1}^{h}\sum_{k=1}^{n}\frac{\beta v_{jk}^2}{1+\beta v_{jk}^2}\right)$$

$$P_2(w,v) = \varepsilon_2\left(\sum_{j=1}^{h}\sum_{i=1}^{m} w_{ij}^2 + \sum_{j=1}^{h}\sum_{k=1}^{n} v_{jk}^2\right)$$

B. Phase II: Feature Fusion

Once the network has learned the features that must exist in summary sentences, we need to discover the trends and relationships among the features that are inherent in the majority of sentences. This is accomplished by the feature fusion phase, which consists of two steps: 1) eliminating uncommon features; and 2) collapsing the effects of common features.

1) Eliminating Uncommon Features

After the training phase, the connections having very small weights can be pruned without affecting the performance of the network. For each input to hidden layer connection (w_{ij}), if $\max_k \left|v_{jk} w_{ij}\right| < 0.1$ remove w_{ij}, and for each hidden to output layer connection (v_{jk}), if $\left|v_{jk}\right| \le 0.1$ remove $v_{jk.}$

As a result, any input or hidden layer neuron having no emanating connections can be safely removed from the network. In addition, any hidden layer neuron having no abutting connections can be removed. This corresponds to eliminating uncommon features from the network.

Once the pruning step is complete, the network is trained with the same dataset in phase one to ensure that the recall accuracy of the network has not diminished significantly. If the recall accuracy of the network drops by more than 2%, the pruned connections and neurons are restored and a stepwise pruning approach is pursued. In the stepwise pruning approach, the incoming and outgoing connections of the hidden layer neurons are pruned and the network is re-trained and tested for recall accuracy, one hidden layer neuron at a time.

2) Collapsing the Effects of Common Features

After pruning the network, the hidden layer activation values for each hidden layer neuron are clustered utilizing an adaptive clustering technique, where G_c is the centroid of cluster *c*.

$$\min\left(Dist\left(G_c, e\right) < r_c\right) \quad \forall c \in U \qquad (4)$$

The clustering algorithm is adaptable, that is, the clusters are created dynamically as activation values are added into the clusterspace. Therefore, the number of clusters and the number of activation values in each cluster are not known *a priori*. The centroid of each cluster represents the mean of the activation values in the cluster and can be used as the representative value of the cluster, while the frequency of each cluster represents the number of activation values in that cluster. By using the centroids of the clusters, each hidden layer neuron has a minimal set of activations. This helps with getting generalized outputs at the output layer. This corresponds to collapsing the effects of common features. In the sentence selection phase, the activation value of each hidden layer neuron is replaced by the centroid of the cluster, which the activation value belongs to. The performance of the network is not compromised, as long as the cluster radius (r_c) is less than the following upper-bound, where the error tolerance, δ, is usually set to a value less than 0.01.

$$\left|r_c\right| \le \frac{\ln\left(\frac{1}{\delta}-1\right)}{\cdot \max_k \left|\sum_{j=1}^{m} v_{jk}\right|} \qquad (5)$$

Since dynamic clustering is order sensitive, the activation values are re-clustered. The radius of new clusters is set to one-half of the original clusters. The benefits of re-clustering are twofold: 1) Due to order sensitivity of dynamic clustering, some of the activation values may be misclassified. Re-clustering alleviates this deficiency by classifying the activation values in appropriate clusters. 2) Re-clustering with one-half of the original radius eliminates any possible overlaps among clusters. The combination of these two steps corresponds to generalizing the effects of sentence features.

Each cluster is identified by its centroid and frequency. Feature fusion phase provides control parameters, which can be used for sentence ranking.

C. Phase III: Sentence Selection

Once the network has been trained, pruned, and generalized, it can be used as a tool to determine whether or not each sentence should be included in the summary. This phase is accomplished by providing control parameters for the desired radius and frequency of hidden layer activation clusters to select highly ranked sentences. The sentence ranking is directly proportional to cluster frequency and inversely proportional to cluster radius. Only sentences that satisfy the required cluster boundary and frequency of all hidden layer neurons are selected as high-ranking summary sentences. The selected sentences posses the common features inherent in the majority of summary sentences.

V. Results and Analysis

We used 85 news articles from the Internet with various topics such as technology, sports, and world news to train the network. Each article consists of 19 to 56 sentences with an average of 34 sentences. The entire set consists of 2,835 sentences. Every sentence, which is represented by a feature vector, is labeled as either a summary sentence or an unimportant sentence. A human reader performed the labeling of the sentences. A total of 763 sentences were labeled as summary sentence with an average of 12 sentences per article. Text summarization process can be applied to both real-valued and binary-valued input vectors. Therefore, we trained three different neural networks, one for real-valued inputs and two for binary-valued inputs.

A. Real-Valued Feature Vectors

For the real-valued feature input, we trained a neural network (N_1) with seven input layer neurons, twelve hidden layer neurons, and one output layer neuron. The input to each input layer neuron, which represents one of seven sentence features, is a real value. The value of the output neuron is either one (summary sentence) or zero (unimportant sentence). The network was trained for 10,000 cycles and achieved a recall accuracy of 99%. After the feature fusion phase, f_1 (Paragraph follows title) and f_4 (First sentence in the paragraph) were removed. In addition, three hidden layer neurons were removed. The removal of f_1 feature is understandable, since most of the articles did not have sub-titles or section headings. Therefore, the only paragraph following a title would be the first paragraph, and this information is already contained in feature f_2 (paragraph location in document). The removal of f_4 feature indicates that the first sentence in the paragraph is not always selected to be included in the summary. We then used the same 85 news articles as a test set for the modified network. The summaries compiled by the network were compared with the summaries compiled by the human reader. The accuracy of the modified network ranged from 90% to 95% with an average accuracy of 93.6% when compared to the desired results obtained from the human reader. The network selected one to five sentences in 30.5% of the articles (26 articles) that were not selected as summary sentences by the human reader. This can be contributed to over-generalization of the network. The network did not select one to three sentences in 12.9% of the articles (11 articles) that were selected by the human reader.

B. Binary-Valued Feature Vectors

For the binary-valued feature input, we discretized the seven sentence features. Each feature is represented as a sequence of binary values. Each binary value represents a range of real values for the sentence feature. We used two different approaches to discretize the real values. In the first approach, we grouped the real numbers into intervals. In the second approach, we discretized the real numbers into single real values.

1) Discretized Real Values into Intervals

Table II describes the discrete intervals for all seven features.

TABLE II
DISCRETE INTERVALS

Feature	Neurons	Intervals
f_1	2	[0],[1]
f_2	4	[1-2],[3-4],[5-9],10+
f_3	4	[1-2],[3-6],[7-9],10+
f_4	2	[0],[1]
f_5	4	[1-4],[5-7],[8-9],10+
f_6	3	[0-4],[5-9],10+
f_7	4	[0-3],[4-6],[7-9],10+

We trained a neural network (N_2) with twenty-three input layer neurons, thirty-five hidden layer neurons, and one output layer neuron. Each feature is represented by a binary vector. For example, f_6 (number of thematic words in the sentence) can be represented as [0 1 0], which implies that there are five to nine thematic words in the sentence. The network was trained for 10,000 cycles and achieved a recall accuracy of 99%. Once again, after the feature fusion phase, f_1 (Paragraph follows title) and f_4 (First sentence in the paragraph) were removed. In addition, seven hidden layer neurons were removed. The accuracy of the modified network ranged from 94% to 97% with an average accuracy of 96.2% when compared to the desired results for the same 85 news articles. The network selected one to three sentences in 14.1% of the articles (12 articles) that were not selected as summary sentences by the human reader, and did not select one to two sentences in 5.8% of the articles (5 articles) that were selected by the human reader.

2) Discretized Real Values into Single Values

We discretized the real numbers up to 10 into individual numbers where each real number is represented by a binary value. In this case, we trained a neural network (N_3) consisting of fifty-four input layer neurons, seventy hidden layer neurons, and one output layer neuron. Each feature, except for f_1 and f_4, is represented by a 10-element binary vector. The network was trained for 10,000 cycles and achieved a recall accuracy of 99%. Once again, after the feature fusion phase, f_1 (Paragraph follows title) and f_4 (First sentence in the paragraph) were removed. In addition, fourteen hidden layer neurons were removed. The accuracy of the modified network ranged from 97% to 99% with an average accuracy of 98.6% when compared to the desired results for the same 85 news articles. The network selected all sentences selected by the human reader. However, for 5.8% of the articles (5 articles), the network selected one to two sentences that were not selected as summary sentences by the human reader. Table III represents the features and their discrete values.

TABLE III
DISCRETE VALUES

Feature	Neurons	Intervals
f_1	2	[0],[1]
f_2	10	[1],[2],[3],[4],[5],[6],[7],[8],[9],10+
f_3	10	[1],[2],[3],[4],[5],[6],[7],[8],[9],10+
f_4	2	[0],[1]
f_5	10	[1],[2],[3],[4],[5],[6],[7],[8],[9],10+
f_6	10	[1],[2],[3],[4],[5],[6],[7],[8],[9],10+
f_7	10	[1],[2],[3],[4],[5],[6],[7],[8],[9],10+

C. Assessing the Accuracy of the Networks

In order to assess the accuracy of all three neural networks, we selected 25 different news articles. The human reader and all three modified networks summarized the 25 news articles, independently. The average accuracy of the *real-valued* neural network (N_1) was 93%, the average accuracy of the *discretized real-values into intervals* neural network (N_2) was 96%, and the average accuracy of the *discretized real-values into single values* neural network (N_3) was 99% when compared with the human reader's summaries.

VI. Conclusions

The performance of the text summarization process depends predominantly on the style of the human reader. The selection of features as well as the selection of summary sentences by the human reader from the training paragraphs play an important role in the performance of the network. The network is trained according to the style of the human reader and to which sentences the human reader deems to be important in a paragraph. This, in fact, is an advantage our approach provides. Individual readers can train the neural network according to their own styles. In addition, the selected features can be modified to reflect the reader's needs and requirements.

References

[1] I. Mani, *Automatic Summarization*, John Benjamins Publishing Company, pp. 129-165, 2001.

[2] W.T. Chuang and J. Yang, "Extracting sentence segments for text summarization: a machine learning approach", *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Athens, Greece, pp. 152-159, 2000.

[3] J. Kupiec, J. Pederson and F. Chen, "A Trainable Document Summarizer", *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Seattle, Washington, pp. 68-73, 1995.

[4] P.B. Baxendale, "Machine-Made Index for Technical Literature: An Experiment" IBM Journal of Research and Development, vol. 2, no. 4, pp. 354-361, 1958.

[5] C.Y. Liu and E. Hovy, "Identifying Topics by Position", *Proceedings of the 5th Conference on Applied Natural Language Processing (ANLP-97)*, Seattle, Washington, pp. 283-290, 1997.

[6] H.P. Luhn, "The Automatic Creation of Literature Abstracts", *IBM Journal for Research and Development*, vol. 2, no.2, pp. 159-165, 1958.

[7] K. Sparck Jones, "A Statistical Interpretation of Team Specificity and its Application in Retrieval", *Journal of Documentation*, vol. 28, no. 1, pp. 11-21, 1972.

[8] H.P. Edmundson, "New Methods in Automatic Extracting", *Journal of the ACM*, vol. 16, no.2, pp. 264-285, 1969.

[9] C.D. Paice, "The Automatic Generation of Literature Abstracts: An Approach Based on Self-Indicating Phrases", *Information Retrieval Research, Proceedings of the Joint ACM/BCS Symposium in Information Storage and Retrieval*, Cambridge, England, pp. 172-191, 1980.

[10] D. Marcu, "From Discourse Structures to Text Summaries", *Proceedings of the ACL/EACL Workshop on Intelligent Scalable Text Summarization*, Madrid, Spain, pp. 256-262, 1997.

[11] R. Brandow, K. Mitze and L. Rau, "Automatic condensation of electronic publications by sentence selection", *Information Processing and Management*, vol. 31, no.5, pp. 675-685, 1995.

[12] M. Porter, "An algorithm for suffix stripping", *Program*, vol. 14, no. 3, pp. 130-137, 1980.