

Chapter 1

Introduction

1.1 Motivation

A technical paper often times comes with presentation slides. Presentation slides of a technical paper are more effective to convey information to readers, especially in academic conferences. However, it is not trivial to generate good presentation slides that concisely summarize the paper and present its key points in an organized and structured way.

We view the problem as automatic text summarization. A set of slides is then a summarized version of the text in the paper. Automatic text summarization is a well-studied area but to the best of our knowledge, very few touch the domain of slides generation. This motivates us to investigate the problem of slide generation.

1.2 System overview

We employ a rule-based approach as a baseline system for slides generation. After manually examined the 20 sets of technical paper and human-written presentation slides, we adopted two heuristic rules. First, the location of sentence is important, and this motivates us to assign a location score for sentences in the paper. Secondly, important phrases are often appeared repeatedly in the paper, and this motivates us to also use keyword-frequency method in our system.

In the second version of system, we improve the baseline system by employing summarization techniques based on corpus statistics. We use an improved HMM model to decompose the slides in testing pairs of slides-documents, obtain training data for machine learning and apply the trained classifier to input technical paper for slides generation.

Since both the heuristic rules employed in baseline system and the classifier training on testing corpus focus on text in slides generation, we ignore the graphs and tables in the technical paper. To simplify the problem we only generate slides with text.

We also conjecture that it would desirable for the user to generate presentation slides at different level of granularity, since users may have different preference. We accommodate such a need by allowing user to choose either the number of slides to be generated (in baseline system) or the number of sentences extracted from the paper (in the second version of the system). The problem then concerns with taking a technical paper and let the user to supply an intended number of slides to generate or sentences to include as input, the system will output a set of presentation slides with desired number of slides or sentences.

1.3 Report outline

In Chapter 2, we will examine related work to our problem, followed by the description of the baseline system in Chapter 3. In Chapter 4, we will give the system architecture and implementation steps for corpus obtaining, machine learning and slides generation in detail. The evaluation results will be presented and analyzed in Chapter 5. We conclude this report by the discussing the feasible further improvement on the reported system.

Chapter 2

Related Work

2.1 Key sentences extraction

We view slides set generation as a text summarization problem. One of the most generic and simple approaches for automatic document summarization is to summarize by sentence extraction. When people are writing presentation slides, they usually look for key sentences in the paper as sources. For the above two reasons, we defined our first step for slides generation as summary by extraction, and examined some previous works in this area.

Automatic sentence extraction is a well-studied area with many different approaches. One way is to look at each sentence for important clues and to compute a total score for the sentence based on those clues. C.D.Paice has examined several distinct clues (C.D.Paice, 1989), namely frequency-keyword, title-keyword, location method, syntactic criteria, indicator phrase, etc. However, it is not trivial to combine different score using simple combination with manually assigned parameters. Either knowledge-based approach or a Bayesian classifier is used to weigh those various clues which may co-occur in one sentence.

Kupiec et.al.(Kupiec et al 1995) develop a trainable document summarizer to perform sentence extraction based on a combination of different heuristics. The system uses a set of five discrete features, namely Sentence Length Cut-off Feature, Fixed-Phrase Features, Paragraph Features, Thematic Word Features and Uppercase Word Features.

A simple Bayesian classifier is trained to assign each sentence a score which can be used to select sentences for inclusion in a summary. The training corpus consists of 188 documents and their abstracts which are created by professional abstractors. In the cross-validation evaluation, for summaries that are 25% of the size of the average test document, it selects 84% of the sentences chosen by professionals. The analysis of performance for individual features shows that the feature gives best individual performance is paragraph feature, recording whether a sentence is paragraph-initial, paragraph-final or paragraph medial. This analysis suggests that position of the sentence might be the most determinant factor in deciding whether it should be included in document extracts.

Our problem of slide set generation is similar to document extraction but different in purpose. The text in presentation slides are targeted at illustrating the key points of the original paper. While the document extracts are required to be as almost as informative as full text of document, the presentation slides may give a higher level overview of the paper and omit some elaborating information. To generate good presentation slides, topic sentences should be given higher priority in sentence extraction process.

Edmundson has introduced four clues for identifying topics, namely title, cue phrases, key words and cue phrases.(Edmundson 1969), . Among these clues, position method remains the best in many studies and analysis. Lin and Hovy (Lin and Hovy, 1997) provide an empirical validation for the position hypothesis that the importance of a sentence in a text is indeed related to its ordinal position. The evaluation used a set of 2907 texts from Vol.2 of the Ziff-Davis corpus, showing that the title plus two most position rewarding sentence provide about 60% of the keywords.

Position of sentence is a determinant factor in both document extraction and topic sentence selection. And this motivates us to start the baseline system with a straightforward position method.

2.2 Sentence reduction

Summary by sentence extraction can identify the key sentences in the document, but the bullet points in presentation slides are not sentences. The purpose of bullet points is to only convey the key information. We view the bullet points as reduced form of extracted sentences.

In some previous summarization system, sentence reduction is a technique applied to improve the quality of summaries generated (Jing, 2000). It automatically removes the extraneous phrases from sentences which are previous extracted from the original document for summarization purpose.

Jing used a corpus consisting of original sentences and their reduced form for training and testing. The corpus was created by a decomposition program using HMM which we will discuss in detail in Chapter 4. She employs the following four steps to reduce sentence: syntactic parsing, grammar checking, context information computing and corpus evidence retrieving. After annotating the sentence parse tree with all these different types of information, a subtree (a phrase) will be removed only if it is not grammatically obligatory, not the focus of local context and has a reasonable probability of being removed by humans. Jing's program has achieved average success rate of 81.3% with 400 training corpus sentences and 100 testing sentences

Jing's method for sentence reduction is mainly a discourse-based approach as it involves parsing the texts and analyzing the context information. Although 500 pairs of sentences and their reduced form are used as training corpus, it only computes the possibility of a specific phrase to be removed in a specific pattern, which can be very sparse and only used as a reference. Her method works well for sentence reduction in general document summarizations, since it mainly bases on generic rules. But as our slide set generation problem is to perform sentence reduction for specific purpose (to

generate bullet points) on a specific domain (extracted key sentences from technical paper), it will be desirable to build statistical model from training corpus rather than directly employ heuristic rules. We need to gather more important feature information in training corpus to approach our problem in a more statistical way.

2.3 Slides alignment and summary decomposition

Slides alignment is essential in slides generation system. The manual alignment indicates the rules people adopt when they prepare their presentation slides. And it is particularly essential to have slide alignment as training corpus to train a statistical model, as in our case.

Ezekiel and Kan (Ezekiel and Kan, 2006) have proposed a method to align each presentation slide to its most similar document paragraph. And this approach achieves highest accuracy rate of 65% in the 20 computer science presentation-document pairs in evaluation process. However, the alignment information between slides and paragraphs fails to provide accurate and adequate knowledge for machine learning on slides generation due to two reasons. First, the alignment system does not take care of multiple and fractional alignments between slides and paragraphs, while in many real situations, people will extract sentences from different paragraphs to generate one slide. Secondly, the alignment process only decomposes the slides to paragraph level, while corpus data for machine learning requires alignment between slides phrases and sentences. We need to further decompose all the words and phrases in the slides to align them to different sentences instead of paragraphs.

Nevertheless, the slides alignment results can still be good reference factors in the later process of further decomposing. Thus, we use the 20 computer science

presentation-document pairs, together with their slides-document alignment information, as training corpus in slides generation system.

We also studied summary decomposition method proposed by Jing, which is to use Hidden Markov Model (HMM) to align sentences in newspaper article with phrases in corresponding summaries. The system determines the most likely position in the document of each word in the summary using a set of heuristic rules. We will use this technique and a more detailed description of this method will be discussed in the Chapter 4.

In evaluation, Jing's method achieves around 42% accuracy for aligning sentences to single sentences. The figure shows that adopting Jing's method directly in slides alignment introduces high alignment errors. The reason is that Jing bases her method on the assumption that all words in the summaries are directly extracted from the original text. This assumption is unrealistic in real situation when some of the words and short phrases might be inserted to the slides, details will be discussed in Chapter 4.

2.4 Summary evaluation method -- ROUGE

ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation. It is a measure of the quality of automatic generated summary by comparing it to summaries written by human. It counts the number of overlapping units such as n-gram, word sequences and word pairs between the summary pairs and introduces different ROUGE measures as ROUGE-N, ROUGE-L, ROUGE-W and ROUGE-S

ROUGE-N is an n-gram recall between a candidate summary and a set of reference

summaries. It is computed as follows:

$$ROUGE - N = \frac{\sum_{S \in \{referenceSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{referenceSummaries\}} \sum_{gram_n \in S} Count(gram_n)}$$

Where n stands for the length of the n-gram, $Count_{match}(gram_n)$ is the maximum number of n-grams co-occurring in a candidate summary and a set of reference summaries. ROUGE-1, ROUGE-2, ROUGE-3 and ROUGE-4 are commonly used measures.

ROUGE-L is a measure of longest common sequence between two summaries. The idea here is that the longer the longest common sequence is, the similar two summaries are. ROUGE-L does not require consecutive matches.

ROUGE-W improves on ROUGE-L by measuring weighted longest common sequence. For a reference words sequence X and two candidate words sequence Y and Z as follow:

X: [A B C D E F G]

Y: [A B C D O P Q]

Z: [A T E B C W D]

Rouge-L cannot differentiate between Y and Z since they have same longest common sequence. But ROUGE-W will choose Y over Z as words sequence in Y are in consecutive match with reference sequence X.

Chapter 3

Baseline System

3.1 System outline

We define our baseline system as a simple rule-based system. In Chapter 2, related work shows that position of sentences is a determinant factor in document extracts generation and topic sentence extraction. It can outperform methods rely on other individual feature, such as key words and cue phrases. Position method is adopted in our baseline system.

Although frequency-keyword method sometimes gives poor individual performance, (Edmundson 1969), we still included it in our system. It helps to differentiate two sentences with same position feature value, and the combination of two methods make the system more robust.

We then use the following two rules for slide set generation in baseline system.

1. Within one section, the first paragraph is more important than the rest. Within one paragraph, the first sentence and the last sentence are more important than the sentences in the middle. (position method)
2. Phrases appear multiple times in the original paper are more likely to be included in the presentation slides. (frequency-keyword method)

In the second rule, Term Frequency * Inverse Document Frequency (tf*idf) value is

assigned for each sentence. We use document frequency for terms with top 30,000 ranking found on 49,602,191 pages on web, which is a joint effort of the UC Berkeley and Stanford Digital Library Projects. The document frequency of these terms are preprocessed and stored in a hashtable. It will be loaded to memory in the running time.

We examined twenty pairs of technical paper and corresponding presentation slides. The ratio of slides number to paragraph number varies significantly. The system then requires users to input intended number of slides to generate.

To simplify the problem at this stage, we make two further assumptions. First, the content of one slide can only be extracted from one paragraph. Secondly, after we identified the key paragraph, we extract L sentences with highest score to generate the slide. And we use $L = 3$ in our experiment.

3.2 Implementation steps

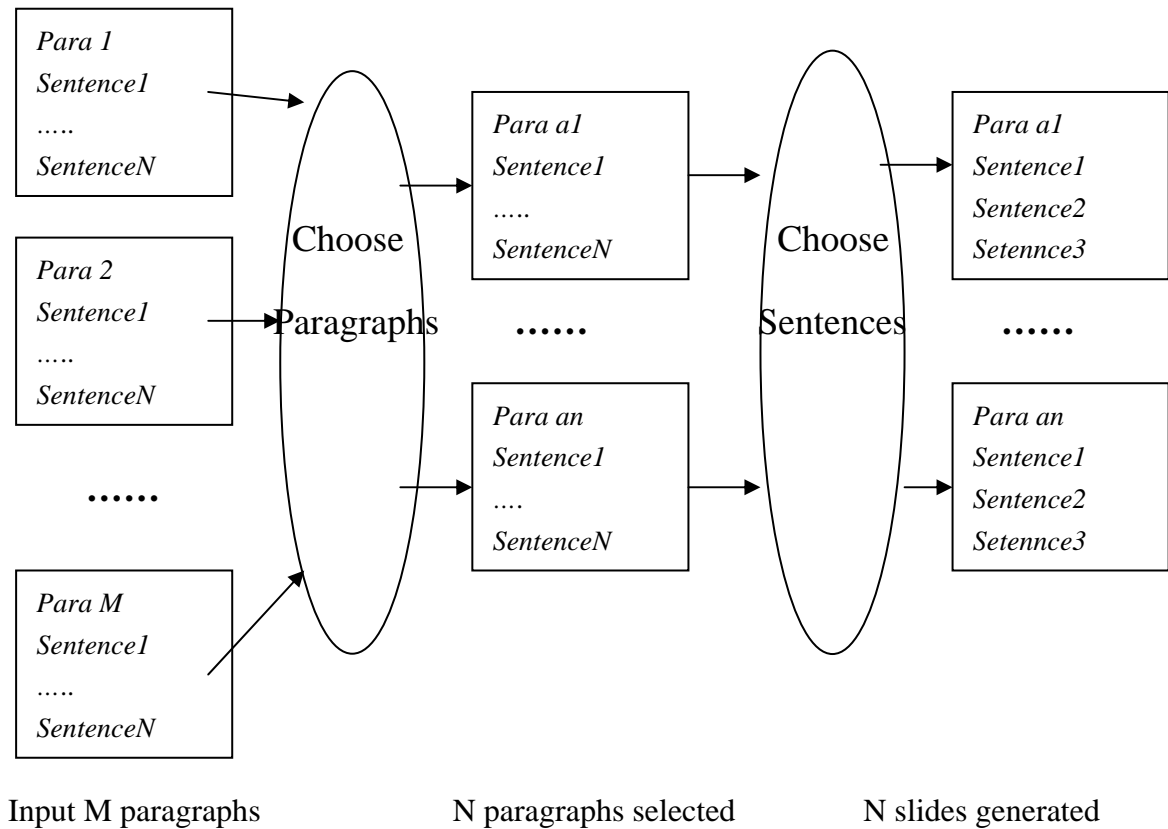


Figure 1 Outline of baseline system

3.2.1 Choosing key paragraphs

The technical paper is converted from pdf to txt format before passing to the system as input. Paragraph number is labeled at the beginning of each paragraph. Each heading is separated as one individual paragraph. The paragraphs in the paper can be classified into three categories: paragraph only contains heading, paragraph below heading and paragraph not directly related to heading. Heading paragraphs are easy to

detect, as they only consist a short phrase (we use 3 as a cut-off length in our experiment), and starts with a section number.

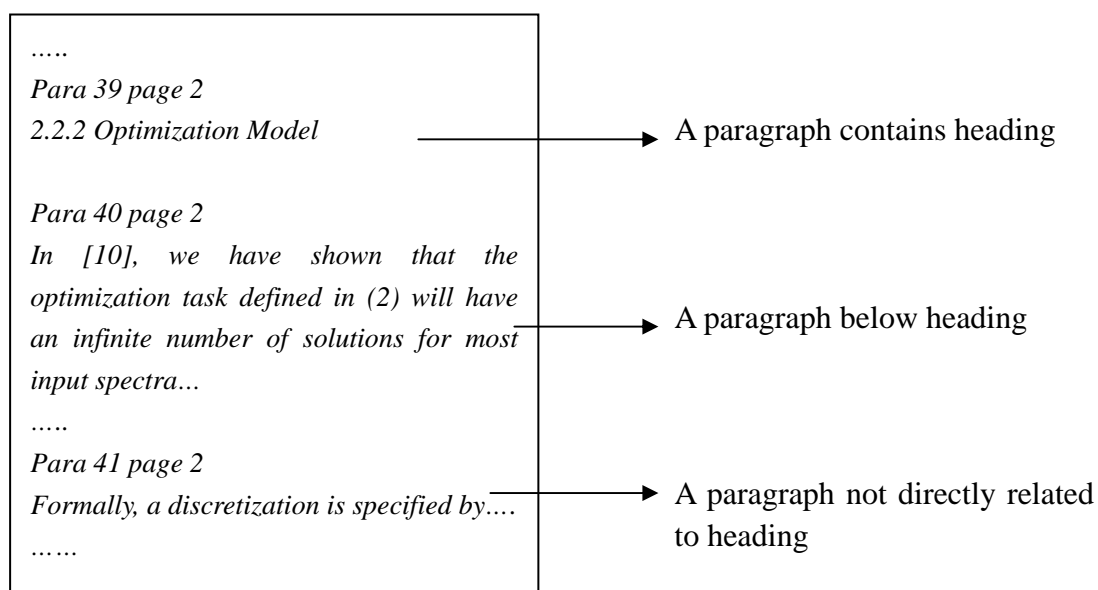


Figure 2 A fragment of paper input to the baseline system

The user is also required to input the number of slides he intends to generate.

For an input paper with M paragraphs, if the number of slides we intend to generate is N , we formalize the problem in the first step as choosing N paragraphs with most important content out of M paragraphs in the full technical paper.

Among all the M paragraphs, we first remove paragraphs which contains less than L sentences ($L = 3$), except for heading paragraphs. Because we make the assumption in the beginning that L sentences will be extracted from key paragraphs to generate one slide, the paragraphs with less than L sentences will not be selected as key paragraphs.

Next step we will select the paragraphs using position method. According to the rules we adopted, the paragraph below heading in the section will be more important than paragraphs not directly related to heading. In the example in Figure 2, paragraph 40

will be preferred over paragraph 41.

Suppose the number of paragraphs below the section headings is K , if K is larger than intended number of slides N , we will choose N paragraphs with highest $tf*idf$ value (average of $tf*idf$ value for all sentences in the paragraph) out of the K paragraphs below the headings.

If K is smaller than the intended number of slides N , it means that $N-K$ paragraphs need to be selected from the rest of the paragraphs in the paper. Again the $N-K$ paragraphs with highest $tf*idf$ value will be selected.

3.2.2 Key sentence extraction

For the N paragraphs selected from the first step of implementation, three sentences with highest scores in each paragraph are going to be extracted to generate the slide.

The two clues we use in sentence scoring process are location score and $tf*idf$ value. The sentence at the beginning and ending of the paragraphs are assigned with highest location value while the sentence in the middle is assigned the lowest value. To give $tf*idf$ value for a sentence, we first remove all the words in stoplist, compute $tf*idf$ for the rest of the words, and use the average value as $tf*idf$ score for the sentence.

We use a simple linear combination of these two factors, each weighs 0.5 in the final sentence score.

3.3 Evaluation

We have run the baseline system on twenty technical papers, each with a set of human

written slides as peers. ROUGE-1 and ROUGE-2 values are computed and will be presented and discussed in Chapter 5.

3.4 Limitations

The baseline system is a simple and subjective system. There are limitations on the performance due to following reasons.

First, only a small number of paragraphs are chosen for sentence extraction due to the assumption that one slide corresponds to only one paragraph. A large percentage of information is lost in this step.

Secondly, the sentence scoring scheme might not be the best as we manually assigned the parameters. Without a training corpus and machine learning technique it is hard to accurately select the correct sentence.

What is more, the slides generated from the baseline system are not in the bullets and phrases form but sentences. Many phrases in the sentence are redundant.

We will target on these weakness and limitations on the final version of the slides generation system.

Chapter 4

System Architecture

The baseline system we discussed in the previous chapter is a handcrafted rule-based system. In our working system we rely mainly on training corpus. We approach slides generation as a statistical classification problem. Given training pairs of technical paper and corresponding human-written presentation slides, we develop a classification function that helps to estimate the probability given sentences in technical paper is included in the slides generated. Furthermore, as bullet points in presentation slides are not whole sentences, we also build a classifier to estimate the probability for words within the sentence to be removed.

The following are the major three steps to generate slides:

1. Use an HMM model to decompose the training set of slides.
2. Compute feature vector representation for each sentence for training a machine learning algorithm and apply it to sentence extraction.
3. Apply sentence reduction to extracted sentences.

4.1 Step 1: Decompose training slides using HMM model

The goal of decomposition problem is to determine the relationship between training set of presentation slides and technical paper. For each word that appears in the slide, the automatic decomposition system decides whether it comes from the original text. If the system decides it does, it also identifies a position in the paper where it most likely originated from.

We have performed decomposition on 20 computer science presentation-document pairs. In each pair, slides are already aligned to the most similar paragraph in the paper. We use the alignment information between slides and paragraphs as reference in the decomposition process.

4.1.1 Training data preprocessing

The training data input to the decomposing system is a pair of technical paper P and corresponding presentation slides V . P is a set of paragraphs, represented as $\{P_1, P_2, P_3, \dots, P_N\}$, and V is a set of slides, represented as $\{V_1, V_2, V_3, \dots, V_m\}$. For each slide, V_x , it can either be aligned to a paragraph P_y or a “nil” if no paragraph it can be aligned to. The alignment of slides V and paper P is also input to the training system.

Technical paper P

Presentation slides V

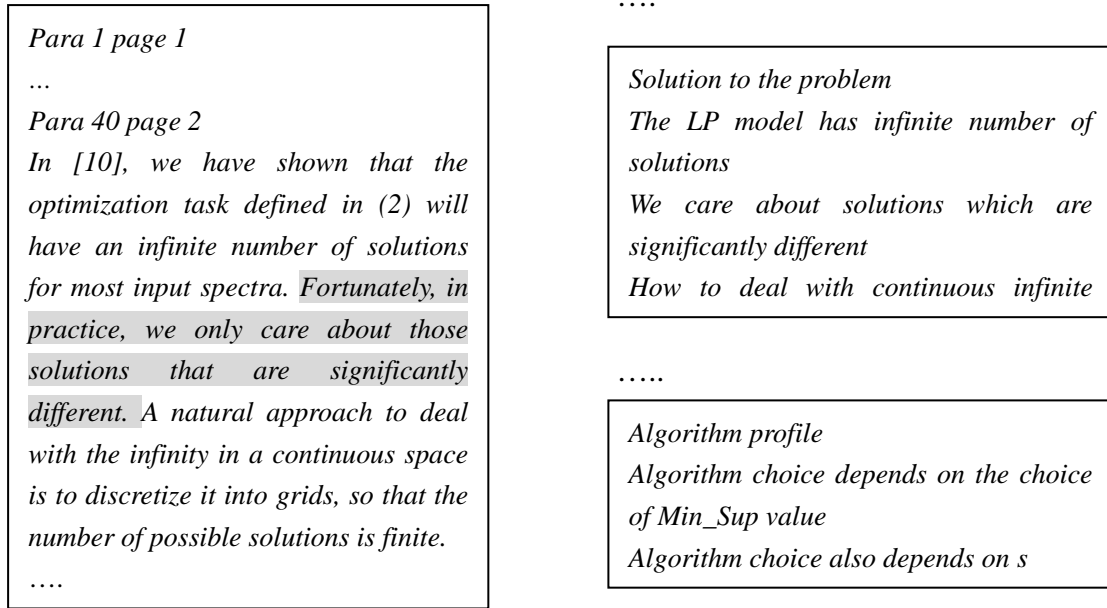


Figure 3 input training set

For each word in presentation slides V, the processing module will find all the possible locations in technical paper P. Each possible location is presented as $(PNUM, SPNUM, SNUM, WNUM)$

<i>PNUM</i>	Paragraph position in the paper
<i>SPNUM</i>	Sentence position within the paragraph
<i>SNUM</i>	Sentence position in the paper
<i>WNUM</i>	Word position within the sentence

Table 1: meanings of names in location fields

In the figure 2 input example, the word “significantly” has a possible location (40, 2, 99, 12).

Multiple occurrences of a word in the slides are represented by a set of word positions $\{(PNUM_1, SPNUM_1, SNUM_1, WNUM_1), (PNUM_2, SPNUM_2, SNUM_2, WNUM_2), \dots, (PNUM_M, SPNUM_M, SNUM_M, WNUM_M)\}$

In the example, the word “that” in the first slide corresponds to a location set{.....
 ..(40, 1, 98, 6), (40, 2, 99, 10), (40, 3, 100, 20)..... }.

4.1.2 Jing’s method

Jing’s method aligns sentences in summary with sentences in original document using an HMM. Her method uses a sentence position and the word’s position within the sentence to present a word position in document. For example, (3,2) refers to the position of the second word in third sentence in the summary. For each word appears in summary, a set of corresponding positions are located.

Jing’s decomposition process is guided by two general heuristic rules: First, human are more likely to cut phrases than single, isolated word; Second, humans are more likely to combine nearby sentences into a single sentence than those far apart.

Then the values of P1-P6 are assigned to transition possibilities between every possible pair of positions, $(SNUM_1, WNUM_1)$ and $(SNUM_2, WNUM_2)$ using the following heuristic rules:

If $((SNUM_1 == SNUM_2) \text{ and } (WNUM_1 = WNUM_2 - 1)) \Rightarrow P1$

If $((SNUM_1 == SNUM_2) \text{ and } (WNUM_1 < WNUM_2 - 1)) \Rightarrow P2$

If $((SNUM_1 == SNUM_2) \text{ and } (WNUM_1 > WNUM_2)) \Rightarrow P3$

If $(SNUM_2 - CONST < SNUM_1 < SNUM_2) \Rightarrow P4$

If $(SNUM_2 < SNUM_1 < SNUM_2 + CONST) \Rightarrow P5$

If $(|SNUM_2 - SNUM_1| \geq CONST) \Rightarrow P6$

Jing’s method uses Viterbi algorithm to find the most likely document position. P1 to P6 are assigned decreased values. P1 is equal to 1 and P6 is a very small number(we

use $P6 = 0.01$ in experiment). *CONST* can be any small constant integer, like 5 or 10.

4.1.3 Improvement of Jing's method

Similar word sequences are used repeatedly in technical paper, there are often more than one candidate for one phrase in the slides. Jing's method doesn't differentiate between these candidates since the transition probabilities are the same in all $P6$ jump. If the word occurs multiple times in the $P6$ region, randomly pick up one location may cause misalignment.

However, with slide-paragraph alignment information available, it is possible to reduce the number of misalignment in this case.

We modified the presentation of location in Jing's work. The location does not only contain the sentence sequence number and word sequence number, but also paragraph number. For each slide that is aligned to a paragraph in the paper, all the words in the slide will keep this aligned paragraph number as a local parameter.

Suppose a word \mathbf{w} in slides has a set of locations in paper represented as $\{(PNUM_1, SPNUM_1, SNUM_1, WNUM_1), (PNUM_2, SPNUM_2, SNUM_2, WNUM_2), \dots, (PNUM_M, SPNUM_M, SNUM_M, WNUM_M)\}$

The word \mathbf{w} will also have a parameter P , which is the paragraph number aligned to the slide \mathbf{w} comes from.

When the transition values to all the possible positions of \mathbf{w} are equal to $P6$, which means that no nearby sentences contains the word \mathbf{w} , we compare the P value with the $PNUM$ value in each positions, and find the $PNUM_n$ which is closest to the

value P . If $|PNUM_n - P| < I$ (I is a small constant integer, we use $I = 3$ in experiment), we will modified the transition probability to position $(PNUM_n, SPNUM_n, SNUM_n, WNUM_n)$ from P6 to P5, in order to give priority for jumping to this position.

Another problem with Jing’s work is that she made the assumption that all the words in slides are cut from the original document, unless the word never appears in the document. However, inserted words are very common in human-written slides, especially the conjunctive expressions or function words. Jing’s method is likely to cause alignment errors in these cases.

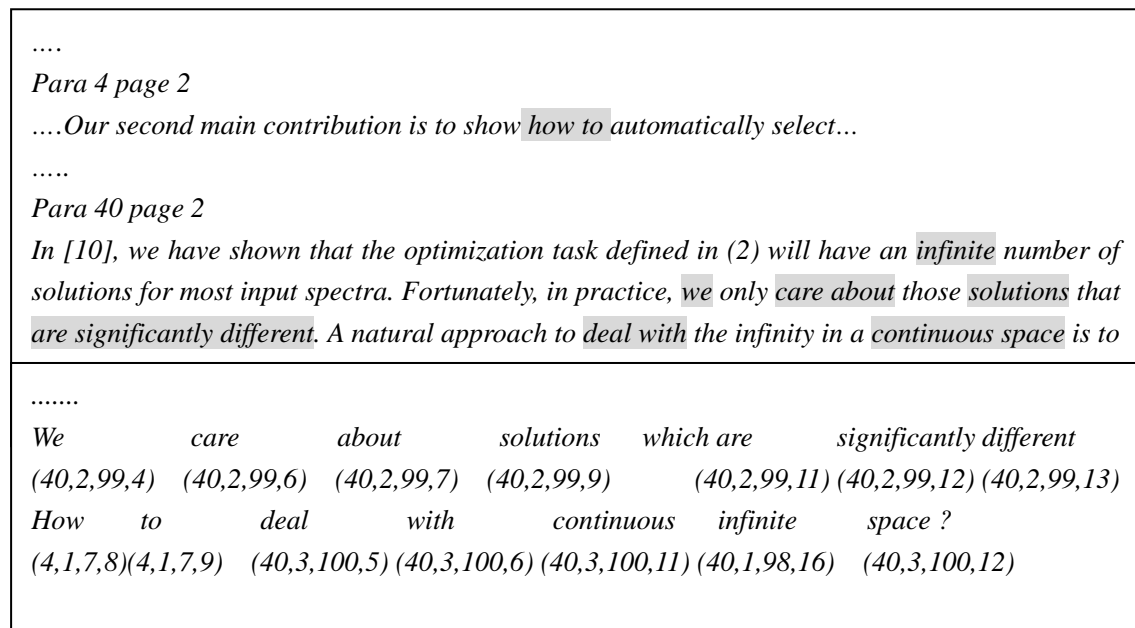


Figure 4. Result of decomposition using Jing’s method. Numbers in parentheses represent aligned word position in paper.

In the above example, the word “how ” is an inserted word, but it is misaligned to the word “how ” in another paragraph in the paper. Since this phrase has appeared in the original paper, Jing’s method cannot classify it as inserted phrase.

We add one more transition possibility, P7, to the existing heuristic rules to solve the

problem. We set the value of P7 equals to P4 in the experiment.

When we move from one location to the next word, other than assigning transition possibilities with value P1-P6 to a set of possible locations, we also add one more position, which is the same as the current location, to the list of possible position for next word, the transition possibility is P7.

The meaning for P7 is that the next word is inserted other than cut from anywhere else, it will not be jumped to other locations but to stay at the current position.

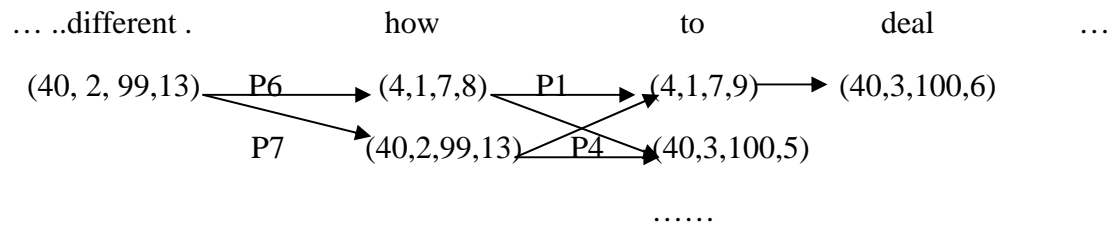


Figure 5 sequence of positions in sentences

In the above example, we add one more transition from the word “different” to “how”, as $P7 * P4$ is larger than $P1 * P6$, although P1 is larger than P4, the sentence will follow the sequence of (40,2,99,13), (40,2,99,13), (40,3,100,5), (40,3,100,6). The word “how” in this sentence will be correctly classified as an inserted word.

4.2 Step 2: Construct feature vector of sentences for machine learning

4.2.1 Features inventory

After we decompose the training slides, find the most likely position for words in the slides, it is trivial to locate extracted sentences in training technical paper.

We decided that if more than N words are cut from one sentence, this sentence will be marked as selected sentence. Otherwise it will be marked as rejected. We perform slides generation on twenty pairs of technical paper and corresponding slides and use $N = 3$ in our testing.

In order to construct a classifier based on training texts, we need to generate a feature vector for each training sentence. Following knowledge source are considered.

Position within the paragraph

Within the paragraph, the beginning and ending sentences are more usually more central to the theme to the text. We classified sentences positions as first sentence, final sentence and middle sentences. If there is only one sentence in the paragraph, it is classified as first sentence.

Paragraph position within the paper

This discrete feature records whether the sentence is within a paragraph below the headings. The paragraphs below headings tend to be more important because they usually summarize the main points in this section.

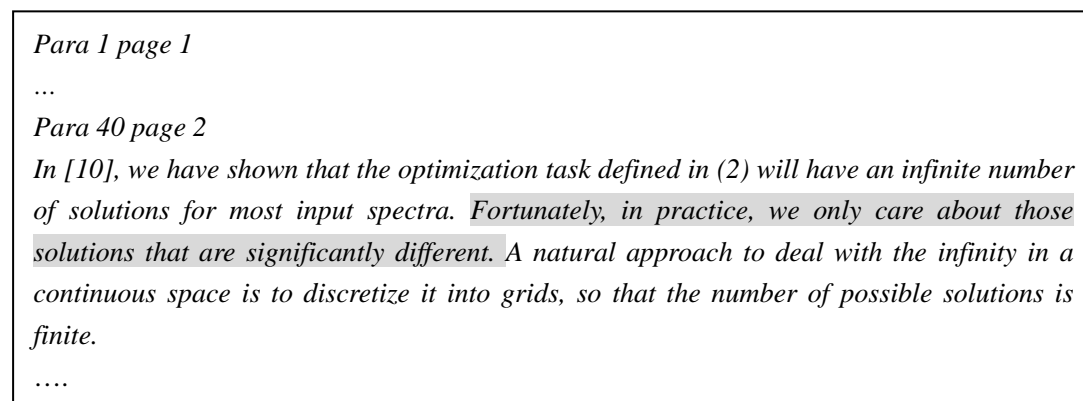
Thematic Word Feature

Sentences (non-significant words have been eliminated by a stoplist) with higher tf*idf value often contain more phrases repeatedly appear in the paper. These sentences are likely to be extracted.

We have computed tf*idf value in baseline system for each sentence. For machine learning purpose, we sorted all the sentences by tf*idf value. The top 10% sentences will receive feature value of 10, next 10% percent will have value of 9, so on and so forth. The last 10% will be assigned feature value of 1. The higher the feature value is, the more thematic words the sentence is likely to contain.

Length of the sentence

Short sentences are less likely to be extracted. We give a threshold, M words. The discrete feature is true for all sentences longer than the threshold, false otherwise. We use $M = 7$ in our system.



Para 1 page 1
...
Para 40 page 2
In [10], we have shown that the optimization task defined in (2) will have an infinite number of solutions for most input spectra. Fortunately, in practice, we only care about those solutions that are significantly different. A natural approach to deal with the infinity in a continuous space is to discretize it into grids, so that the number of possible solutions is finite.
....

Figure 6 a paragraph from paper

We have all together four attributes in the feature vector. The testing data, with its feature value and classification class, is written to the testing file. For example, the sentence in brown rectangle in Figure 4 will be represented as (middle-sentence, true, 7, true, selected)

4.2.2 Machine learning and data mining

After assigned feature vector for each sentence in the training corpus, we construct a classifier and apply it to all the sentences in the technical paper.

We use Weka for machine learning task in our program. Weka is a collection of machine learning algorithm for data mining tasks. Among all the classifiers provided in the package, we choose to use tree.J48, which is a clone of the C4.5 decision tree learner.

We apply the classifier to all the sentences in input technical paper. The classifier will return predicted class value together with the possibility distributions.

As presentation slides can be generated at different granularity, we allow user to choose the number of sentences to extract at this level, or percentage of total number of sentences. The sentences with top N possibility value on class “selected” will be extracted for slides generation in later stage.

4.3 Step 3: Sentence reduction

In most presentation slides text are in bullet point form. It means that some extraneous phrases will be removed from the extracted sentences. It is desirable to implement an automatic sentence reduction system, which is targeted at improving the conciseness of summarization.

4.3.1 Features selection and machine learning

The training corpus we used to build the sentence extraction classifier can also be used to train a classifier on words selection. We analyze each word in the training sentences, construct a feature vector and use WEKA to train a classifier.

The following knowledge sources are studied. We use one sentence in the slide and aligned sentence in the paper as example.

....
Para 40 page 2
In [10], we have shown that the optimization task defined in (2) will have an infinite number of solutions for most input spectra. Fortunately, in practice, we only care about those solutions that are significantly different. A natural approach to deal with the infinity in a continuous space is to discretize it into grids, so that the number of possible solutions is finite.

Figure 6 paragraph from paper, the content in gray rectangular are selected in the

Length of continuous phrase

The word appear in longer phrases are more likely to be selected in the presentation slides, as humans are more tend to cut a phrase from a sentence instead of pick up one single word to generate slides. It could be an important feature to decide whether the

word will be moved from the sentence.

We use the length of continuous phrase cut from the sentence the word is in. For example, the word “significantly” will have the feature value of 3, since it is in a continuous phrase “are significantly different”. The length of the phrase is 3.

Term Frequency * Inverse Document Frequency

The key word frequency value is another important indication in selection of word. phrases which appear in original paper for multiple times will be more likely to be used in presentation slides. We use $tf*idf$ value as a feature value for each word.

Part-of-Speech tag

For each word, we use a feature vector to encode the knowledge source:

$\langle p_{-2}, p_{-1}, p_0, p_1, p_2 \rangle$. $P_{-i}(P_i)$ is the POS of the i th token to the left(right) of the word, P_0 is the POS of word. A token can be a word or a punctuation symbol. If it a null token, ε is assigned instead of POS.

Chapter 5

Evaluation and Discussion

We evaluate and discuss our work in two aspects, decomposition results from training corpus and slides generated from the system. For generated set of slides, we compared the performance of system using training corpus and machine learning technique with the baseline system.

Dataset

Our training dataset is a corpus of 20 presentation-document pairs mostly drawn from the database community's SIGMOD meetings of 2004 and 2005. It has been used for slides alignment training and testing. (Ezekiel and Kan, 2006). The training dataset also contains the alignment information between presentation slides and paper paragraphs.

5.1 Decomposition results

We use a modified Hidden Markov Model (HMM) for sentences decomposition. In twenty sets of training corpus, an average of 81.7% of the words in presentation slides can be found in corresponding paper. The result shows that theoretically, extracting words from original document can generate slides which are able to cover the key points of the paper.

By adding an insertion transition possibility P_7 to HMM, we reduced the number of aligned words by 2.2%. These words are considered as inserted rather than cut from the original document. In our experiment, we set value of P_7 to be equal to P_5 . The P_1

to P6 value we use are 1.0, 0.9, 0.8, 0.3, 0.2, and 0.01.

If we assume that a sentence in document is an aligned sentence when at least one word in the corresponding set of slides is considered as originated from this sentence, then the ratio of aligned words in slides to total number of aligned sentences in paper ranges from 2.3 to 4.1 in twenty slides-document pairs. Thus we use number of 3 as a cutoff point. Among all the words in slides, an average of 72% of total aligned words align to sentences which at least have three words in slides aligned to it. This is a good indication that most words in presentation slides are cut from comparatively small number of sentences in the original paper. They are clustered in some key sentences rather than dispersed all over the paper.

The analysis of decomposition results illustrates the importance of sentence extraction in slides generation. We mark sentences in paper with more than three words appear in slides as selected sentences, assign feature vectors and use them in classifier training, as we discussed in the Chapter 4.

5.2 Presentation slides

Baseline system allows users to input the number of slides they intend to generate. In order to compare generated slides with peer slides, we choose N to be the same as the number of slides written by human.

We compute ROUGE-1 and ROUGE-2 values for the slides generated by baseline system. The average ROUGE-1 recall value is 0.52 and the precision value is 0.32, which means that in average, the slides generated by baseline system contains 52% of the content in targeted human written slides, and only 32% of content in slides

generated are targeted content.

Although average precision value is smaller than recall value, in individual measurement between two sets of slides, the precision value might be higher than recall value. It largely depends on number of slides and level of abstraction in human written summaries. In the twenty sets of presentation slides and documents, the ratio of paragraphs numbers to slides numbers ranges from 2.71 (220 paragraphs to generate 81 slides) to 25.56(406 paragraphs to generate 16 slides). As sentence number per slide is defined in baseline system, the more sentences a human written slide contains, the more possible that the precision value to be larger than the recall value.

In our second version of system, in order to compare with baseline system, we pick up same number of sentences as in the slides generated by baseline system. The ROUGE-1 recall value has been improved to an average of 0.71, and an average precision value of 0.38. The selection of the sentences is no longer limited to specific number of paragraphs, this contributes to the improvement of recall value. However the precision value is still low. It might be helpful if fewer sentences are extracted from the documents. Applying sentence reduction also improves precision value, since extraneous phrases are removed from sentences in targeted human written slides.

Chapter 6

Conclusion and Future Work

In this project, we first build up a baseline system for slides generation using a combination of position method and keyword frequency method, and also improve the baseline system by employing summarization techniques based on corpus statistics. We use an improved HMM model to decompose the slides in testing pairs of slides-documents, obtain training data for machine learning and apply the trained classifier to input technical paper for slides generation.

We can further improve the system by analyzing more sentence reduction and combination techniques. In the future, we may explore on the structure of paragraphs, and combine the discourse method with statistical model, to present slides in a more organized and structured way.

Reference

- C.D.Paice (1990) Constructing literature abstracts by computer: Techniques and prospects. *Information Processing and Management*, 26:171-186,1990.
- Ezekiel E.Ezekiel and Min-Yen Kan (2006) : Methods for Presentation to Document Alignment.
- Julian Kupiec, Jan Pedersen and Francine Chen (1995) A Trainable Document Summarizer
- Lin, Chin-Yew and E.H.Hovy. (1997) Identifying Topics by Position. *In the Proceeding of the 5th Conference on Applied natural Language Processing (ANLP97). Washington, D.C.*
- Lin, Chin-Yew.(2004) ROUGE: a Package for Automatic Evaluation of Summaries. *In proceedings of the Workshop on Text Summarization Branches Out (WAS 2004), Barcelona, Spain*
- H.P.Edmundson (1969). New methods in automatic extracting. *Journal of the ACM*, 16(2):264-285
- Hongyan Jing and Kathleen R.McKeown.(1999). The decomposition of human-written summary sentences. *In Proceeding of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics, volumn 1*
- Hongyan Jing and Kathleen R.McKeown (2000). Cut and paste based text summarization. *In Proceedings of NAACL2000*
- Hongyan Jing (2000) Sentence reduction for automatic text summarization. *In Proceedings of ANLP 2000.*
- P.B.Baxendale. (1958) Machine-made index for technical literature-an experiment. *IBM Journal, pages 354-361, October*