A Document Graph Based Query Focused Multi-Document Summarizer

By

Sibabrata Paladhi

and

Dr. Sivaji Bandyopadhyay

Department of Computer Science and Engineering

Jadavpur University

Jadavpur, Kolkata

India
• **Introduction**
• Graph Based Modeling
• Building Query Independent Components
• Building Query Dependent Components
• Summary Generation
• Post Processing
• Results and Discussion
• Multi-Lingual Scenario
• Conclusion
INTRODUCTION

- Netizens are relying on the Internet for daily news
- Almost 100 newswires available in Internet nowadays
- A google search results in numerous links for a news event
- Web Search engines generally produce query dependent snippets
- A query dependent multi-document summarizer generating query based short summary of news articles will be helpful
Multi-Document Summarization System (MDS)

- **Input**: Multiple source documents about a single topic or an event
- **Output**: Summarized information content using much less lengthy text

**Application:**
- News portal presenting summarized article from different sources
- Corporate emails organized by subjects
- Medical reports about a patient

**Classification Of MDS:**
- Query Independent, Query Dependent
- Sentence Extraction Based, Sentence Fusion Based
• Introduction

• **Graph Based Modeling**
  • Building Query Independent Components
  • Building Query Dependent Components
  • Summary Generation
  • Post Processing
  • Results and Discussion
  • Multi-Lingual Scenario
  • Conclusion
Graph based Modeling

- Nodes are text fragments (Paragraphs in our case)
- Edges reflect relation between nodes
- Edge Scores are correlation measure between corresponding nodes

Software major IBM has developed a speech recognition technology in Hindi which would help physically challenged and less literate Hindi speakers access information through a variety of applications.

The Desktop Hindi Speech Recognition Technology developed by the IBM India Software Lab in collaboration with Centre for Development of Advanced Computing would provide a natural interface for human-computer interaction.

A spellchecker to correct spoken-word errors also enhances the accuracy.

"IBM believes in taking high-end research to the benefit of the masses and bridging the digital divide through a faster diffusion process", Dias said.

The technology also would enable C-DAC to ensure high-level accuracy in Hindi translation in a host of domains, including administration, finance, agriculture and small scale industry.
Steps of Algorithm:

- Perform Query Independent Preprocessing
  - Identification of Seed Nodes
  - Offline document graph Construction

- Perform Query Dependent Processing
  - Evaluation of nodes w.r.to query set
  - Expansion of offline graph so that all keywords are satisfied (AND Semantic)

- Construction of (Best) spanning tree constituting relevant nodes
- Post Processing (Text Ordering)
- Output Summary
OUTLINE

- Introduction
- Graph Based Modeling
  - **Building Query Independent Components**
  - Building Query Dependent Components
  - Illustration of Algorithm
- Post Processing
- Results and Discussion
  - Multi-Lingual Scenario
- Conclusion
Building Query Independent Components

How to find seed/topic nodes?

- Assumption: Paragraphs of $d_i$ are relevant if they are highly related to the relevant paragraphs of $d_j$
- Consider Document set having two related documents
- Construct a bipartite graph $G_b$ across the document set
- Find ‘total edge score’ of individual nodes of $G_b$
- Select nodes having ‘total edge score’ (> threshold) as topic/seed nodes

Major Challenge is to identify information overlap
Offline Construction of Search Graph

- A Document graph $G_D$ is constructed with seed/topic nodes.
- Edge scores are calculated using following formula:

  \[
  E\text{Score}(e) = \frac{\sum_{w \in (t(u) \cap t(v))} \left( tf(t(u), w) + tf(t(v), w) \cdot idf(w) \right)}{size(t(u)) + size(t(v))}
  \]

  - $tf(d, w) = \text{no. of occurrence of } w \text{ in } d$
  - $idf(w) = \text{inverse of no. of documents containing } w$
  - $size(d) = \text{size of } d$
- Only edges having scores > threshold are considered.

Note: Lexicon and Stemmer must be used to find equivalent words.
P0: Software giant IBM has developed a speech recognition software in Hindi.
P1: The company hopes that this development will help physically challenged and less literate Hindi speakers to access information using a variety of applications.
P2: The Desktop Hindi Speech Recognition Technology developed by the IBM India Software Lab in collaboration with Centre for Development of Advanced Computing would provide a natural interface for human-computer interaction.
P3: The new IBM technology could help to provide a natural interface for human-computer interaction.
P4: According to Dr. Daniel Dias, Director, IBM India Research Laboratory, the technology which helps transcribe continuous Hindi speech instantly into text form, could find use in a variety of applications like voice-enabled ATMs, car navigation systems, banking, telecom, railways and airlines.
P5: Besides, the technology could also enable C-DAC to ensure a high level of accuracy in Hindi translation in a number of domains like administration, finance, agriculture and the small-scale industry.
P6: The IBM Desktop Hindi Speech Recognition software is capable of recognizing over 75,000 Hindi words with dialectical variations, providing an accuracy of 90 to 95%.
P7: What’s more, this software also has an integrated spellchecker that corrects spoken-word errors, enhancing the accuracy to a great extent.
P8: The Desktop Hindi Speech Recognition Technology also integrates a number of user-friendly features such as the facility to convert text to digits and decimals, date and currency format, and into fonts which could be imported to any Windows-based application.
P9: "IBM believes in taking high-end research to the benefit of the masses and bridging the digital divide through a faster diffusion process," concluded Dias.

Figure 1. Paragraphs of two news articles with five extracted seed/ topic paragraphs (in bold). Underlined paragraphs are added later during graph expansion phase.
A bipartite graph representing correlation of documents

- Software major IBM has developed a speech recognition technology in Hindi.....
- Called the Desktop Hindi Speech Recognition technology, this software...
- "IBM believes in taking high end research to the benefit of masses and...."
- The technology also would enable CDAC to ensure ...
- Software giant IBM has developed a Hindi speech ...
- The company hopes that this development will help physically challenged ...
- What's more; this also has an integrated spell-checker that corrects spoken word....
- The Desktop Hindi Speech Recognition software integrates a number of user....
OUTLINE

- Introduction
- Graph Based Modeling
- Building Query Independent Components
- **Building Query Dependent Components**
- Summary Generation
- Post Processing
- Results and Discussion
- Multi-Lingual Scenario
- Conclusion
Building Query Dependent Components

- Query based evaluation of $G_D$
  - Nodes of offline document graph are given query dependent score
  - Score reflects relevance of the node (paragraph) w.r.to the keywords
  - Node score is calculated using IR based okapi formula

\[
\sum_{t \in Q,d} \ln \frac{N - df + 0.5}{df + 0.5} \cdot \frac{(k1 + 1) \cdot tf}{(k1(1 - b) + b \frac{dl}{avdl}) + tf} \cdot \frac{(k3 + 1) \cdot qtf}{k3 + qtf}
\]

- $tf = \text{term frequency in node}$
- $qtf = \text{term’s frequency in query}$
- $df = \text{no. of nodes containing the term}$
- $dl = \text{node length (in words)}$
- $N = \text{total no. of nodes}$
- $K1, b, k3$ are constants

Note: Lexicon and Stemmer must be used to find equivalent words.
Partitioning of Query/Keywords Set Q

- $Q_s = \{ q | q \text{ is satisfied by at least one node of } G_D \}$
- $Q_u = \{ q | q \text{ is not satisfied by any node } G_D \}$
- $G_D : \text{Offline Document Graph}$

Falling back to original document set

- Identification of relevant nodes from document set satisfying $Q_u$

Expansion of the offline search graph to include new nodes
P0: Software major IBM has developed a speech recognition technology in Hindi which would help physically challenged and less literate Hindi speakers access information through a variety of systems.
P1: Called the Desktop Hindi Speech Recognition technology, this software was developed by the IBM India Software Lab jointly with the Centre for Development of Advanced Computing.
P2: The technology, which helps transcribe continuous Hindi speech instantly into text form, could find use in a variety of applications like voice-enabled ATMs, car navigation systems, banking, telecom, railways and airlines, said Dr Daniel Dias, Director, IBM India Research Laboratory.
P3: The system can recognize more than 75,000 Hindi words with dialectical variations, providing an accuracy level of 90-95 per cent, he said.
P4: A spellchecker to correct spoken-word errors also enhances the accuracy of the system.
P5: The technology also has integrated many user-friendly features such as facility to convert text to digits and decimals, date and currency format, and into fonts which could be imported to any Windows-based application.
P6: "IBM believes in taking high-end research to the benefit of the masses and bridging the digital divide through a faster diffusion process", Dias said.
P7: The technology also would enable C-DAC to ensure high-level accuracy in Hindi translation in a host of domains, including administration, finance, agriculture and small scale industry.
P8: The IBM Desktop Hindi Speech Recognition software is capable of recognizing over 75,000 Hindi words with dialectical variations, providing an accuracy of 90 to 95%.
P9: What’s more, this software also has an integrated spellchecker that corrects spoken-word errors, enhancing the accuracy to a great extent.
P0: The Desktop Hindi Speech Recognition Technology also integrates a number of user-friendly features such as the facility to convert text to digits and decimals, date and currency format, and into fonts which could be imported to any Windows-based application.
P1: "IBM believes in taking high-end research to the benefit of the masses and bridging the digital divide through a faster diffusion process,” concluded Dias.

Figure 1. Paragraphs of two news articles with five extracted seed/ topic paragraphs (in bold). Underlined paragraphs are added later during graph expansion phase.
Example

- Let keyword set = \{IBM, speech recognition, accuracy\}

- Qs = \{IBM, speech recognition\} and Qu = \{accuracy\}

- Following nodes are identified as relevant w.r.to Qu = \{accuracy\}
  
  1. The system can recognize… **accuracy** level of 90 – 95 ….
  2. A spellchecker to correct….. enhance the **accuracy** ….
  3. What’s more … enhancing the **accuracy** .. .. to a great extent.
Software major IBM has developed a speech recognition system.

The technology also has an integrated user-friendly interface.

The technology, which helps transcribe continuous speech.

The Desktop Hindi Speech recognition technology.

Besides, the technology could also enable CDAC to...

The system can recognize...

accuracy

A spellchecker to correct...

enhance the accuracy

What's more... enhancing the accuracy...

to a
OUTLINE

- Introduction
- Graph Based Modeling
- Building Query Independent Components
- Building Query Dependent Components
- **Summary Generation**
- Post Processing
- Results and Discussion
- Multi-Lingual Scenario
- Conclusion
Summary Generation

- Basically, keyword Search in Expanded Document Search Graph $G_D$
  - Search for Spanning Tree $T_S$ over $G_D$ to cover $Q$
  - Possibility of Multiple Spanning trees satisfying $Q$
  - Selection is done through ranking over $\{T_{\text{Score}}\}$

\[
T_{\text{score}} = a \sum_{e \in T} \frac{1}{e_{\text{score}}} + b \sum_{n \in T} \frac{1}{n_{\text{score}}}
\]

$T_{\text{score}} = \text{Score of individual } T_S \quad e_{\text{score}} = \text{edge score} \quad n_{\text{score}} = \text{node score}$

$a, b$ are constants in the range $[0-1]$

Note: Spanning Tree generation Algorithm can be modified to take into account size requirement of user
P0: Software giant IBM has developed a speech recognition technology in Hindi which would help physically challenged and less literate Hindi speakers access information through a variety of systems.

P1: Called the Desktop Hindi Speech Recognition technology, this software was developed by the IBM India Software Lab jointly with the Centre for Development of Advanced Computing.

P2: The technology, which helps transcribe continuous Hindi speech instantly into text form, could find use in a variety of applications like voice-enabled ATMs, car navigation systems, banking, telecom, railways and airlines, said Dr Daniel Dias, Director, IBM India Research Laboratory.

P3: The system can recognize more than 75,000 Hindi words with dialectical variations, providing an accuracy level of 90-95 per cent, he said.

P4: A spellchecker to correct spoken-word errors also enhances the accuracy of the system.

P5: The technology also has integrated many user-friendly features such as facility to convert text to digits and decimals, date and currency format, and into fonts which could be imported to any windows-based application.

P6: "IBM believes in taking high-end research to the benefit of the masses and bridging the digital divide through a faster diffusion process", Dias said.

P7: The technology also would enable C-DAC to ensure high-level accuracy in Hindi translation in a host of domains, including administration, finance, agriculture and small-scale industry.

P8: The IBM Desktop Hindi Speech Recognition software is capable of recognizing over 75,000 Hindi words with dialectical variations, providing an accuracy of 90 to 95%.

P9: What’s more; this software also has an integrated spell-checker that corrects spoken-word errors, enhancing the accuracy to a great extent.

P0: The Desktop Hindi Speech Recognition Technology also integrates a number of user-friendly features such as the facility to convert text to digits and decimals, date and currency format, and into fonts which could be imported to any Windows-based application.

P1: "IBM believes in taking high-end research to the benefit of the masses and bridging the digital divide through a faster diffusion process," concluded Dias.
Software major IBM has developed a speech recognition technology.

The technology also has an integrated many user-friendly features.

The Desktop-Hindi Speech recognition technology can be used for transcription.

Besides, the technology can also enable users to use voice commands.

A spellchecker to correct errors and enhance accuracy.

What's more, enhancing the accuracy to a higher level.
OUTLINE

• Introduction
• Graph Based Modeling
• Building Query Independent Components
• Building Query Dependent Components
• Summary Generation
• Post Processing
• Results and Discussion
• Multi-Lingual Scenario
• Conclusion
Paragraph Ordering

- Paragraph/Node ordering is a major issue
  - In Single document scenario ordering based on positional rank in original document
  - In Multi-Document original ordering is of no use

- General Techniques
  - Based on Chronological ordering
  - Based on positional rank in original document

- We introduced *Coherency Based Ordering* scheme
  - Select First node based on some heuristics
  - Select $p^{th}$ node which is most coherent to $(p-1)^{th}$ node
Software major IBM has developed a speech recognition tool.

The technology also has an integrated many user-friendly features.

Besides, the technology could also enable voice-to-text conversion.

The IBM Desktop Hindi Speech Recognition technology.

A spellchecker to correct mistakes and enhance the accuracy.

What's more, enhancing the accuracy to a higher level.
D1P0: Software major IBM has developed a speech recognition technology in Hindi which would help physically challenged and less literate Hindi speakers access information through a variety of systems.

D2P6: The IBM Desktop Hindi Speech Recognition software is capable of recognizing over 75,000 Hindi words with dialectical variations, providing an accuracy of 90 to 95%.

D1P4: A spellchecker to correct spoken-word errors also enhances the accuracy.

Ordered Summary for query {IBM, speech recognition, accuracy}
OUTLINE

- Introduction
- Graph Based Modeling
- Building Query Independent Components
- Building Query Dependent Components
- Summary Generation
- Post Processing
- Results and Discussion
- Multi-Lingual Scenario
- Conclusion
Evaluation

- **Types of Evaluation Scheme:**
  - **Extrinsic**: Based on IR task
  - **Intrinsic**: Based on user judgement

- **Quality measured as percentage overlap between system output and manual extract**

- **Quality Parameters**
  - $O_{\text{measure}}$: Percentage overlap with extract having maximum commonality
  - $P_{\text{measure}}$: Percentage overlap with extract having minimum commonality
  - $I_{\text{measure}}$: Percentage overlap with intersection of manual extracts
  - $U_{\text{measure}}$: Percentage overlap with union of manual extracts
Result

Comparison of Different Scores
OUTLINE

- Introduction
- Graph Based Modeling
- Building Query Independent Components
- Building Query Dependent Components
- Summary Generation
- Post Processing
- Results and Discussion
- Multi-Lingual Scenario
- Conclusion
The system can be ported to non-English environment

Requirement of some language dependent tools like

- List of ‘Stop Word’ for that language
- Stemmer / Morphological Analyzer
- Lexicon
OUTLINE

• Introduction
• Graph Based Modeling
• Building Query Independent Components
• Building Query Dependent Components
• Summary Generation
• Post Processing
• Results and Discussion
• Multi-Lingual Scenario
• Conclusion
Conclusion

- Computation is divided into Offline and Online Task
- Applicable as Internet / Intranet based News Summarizer
- The system can be adapted to work in non-English language also !!!
- Performance greatly depends on Selection of Topic nodes
- Summary generation of custom size is possible
- System can be adapted to work as query independent summarizer
Thank You 😊