PROJECT PRESENTATION: TURKISH TEXT SUMMARIZATION

GÜNDÜZ VEHBI DEMIRCI
EMIR GÜLÜMSER

OUTLINE

- Introduction
- Motivation
- Methodology
  - Zemberek
  - Rouge
  - Key Phrases and KEA Algotihm
  - Computational Work
- Expected Results

INTRODUCTION

- In past, retrieving information from a subject was hard due to lack of information or difficulty of finding relevant resources
- With the widespread usage of the internet, documents and resources are brought to online which become accessible for everyone.
- Getting relevant information from this huge amount of data is essential and popular nowadays.
- In this project, we propose a summarization method that extracts Turkish sentences by ranking these sentences using feature calculation.

MOTIVATION

- Today available text search engines return too much documents for a person to identify which one are relevant to his/her needs.
- Presenting document summaries will help people to find their desired documents easily.
- Therefore there is need for technologies that help people on this purpose.

KEY PHRASES

- As mentioned earlier, number of documents on the internet and libraries are increasing.
- From the readers' perspective, finding the required documents becomes a problem as the number of documents increased.
- In order to ease readers work, documents should include key phrases.
- Manually reading, understanding and putting key phrases are very exhaustive and hard work.

KEY PHRASES

- Key phrases can be used for different purposes:
  - in the first page of the academic paper, its aim is to summarization.
  - It is used for indexing in the index part of the paper
  - It can also be used for key words in the search engines in
  - key phrases can be used for different purposes; summarizing, indexing, labeling, categorizing, clustering, highlighting, browsing, and searching

ZEMBEREK

- Zemberek is an open source, general purpose Natural Language Processing library and toolset designed for Turkish.
- Zemberek is officially used as spell checker in Open Office Turkish version and Turkish national Linux Distribution Pardus.
- We use Zemberek for stemming of the Turkish words.

ROUGE

- Recall-Oriented Understudy for Gisting Evaluation
- It is a set of metrics and a software package used for evaluating automatic summarization
- The metrics compare an automatically produced summary or translation against a reference or a set of references (human-produced) summary or translation.

METHODOLOGY

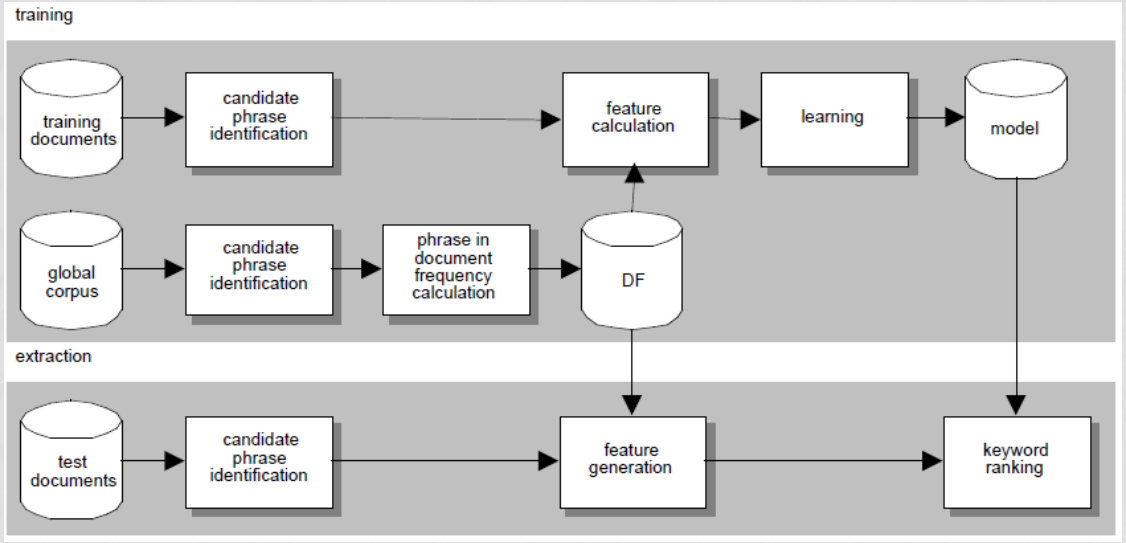
- First, we clean the text using stop word list and stemming.
- We use Zemberek for stemming of the Turkish words.
- We evaluate the words for candidate sentences using feature calculation techniques such as TFxIDF calculation.
- We use training data in the extraction stage.

METHODOLOGY

- Finally, in the extraction stage using training data and the feature values, candidate words are extracted
- First we evaluate the sentences in word / phrase base and calculate their feature values using of Automatic Keyphrase Extractor Algorithm (KEA)
- Using these values we rank the sentences and extract the desired number of sentences to be used for the document's summary.
- In the last step we will compare automatically produced summaries by our algorithm with human-produced summaries.

COMPUTATIONAL WORK

- Uses most of the stages of Key Phrase Extraction Algorithm (KEA).
- Consist of 3 main stages.
  - Training,
  - Extraction,
  - Sentence ranking stages.



TRAINING STAGE

- Mainly, reads the training data consist of several documents with author assigned key phrases.
- Creates a model to be used in extraction stage.
- Firstly, stop words are loaded (acaba,altmış,altı,ama ...)
- Training files are loaded to the system (users can select how many to be loaded).
- The sentences / blocks are split into tokens.

SELECTING CANDIDATE KEYPHRASES

- In the input cleaning part
  - All the punctiation marks are replaced with selected letter (like @).
  - Aphostrophes and no letter tokens are removed.
  - Acronyms are handled as single token.
- In the phrase identification part,
  - In the original algorithm candidate phrases are limited to certain length (i.e. Keyphrase can consist of 2 or more words) .
  - For sentence evaluation, we need one word length phrases.
  - Candidate phrases are cleaned from stop words.

SELECTING CANDIDATE KEYPHRASES

- Case fold and stemming part,
  - To ease the calculations all the phrases are lower cased.
  - Using Zemberek, all the phrases are stemmed (selected the minimum length stems - roots).
- In the feature calculation part,

$$\text{TF}\times\text{IDF} = \frac{\text{freq}(P, D)}{\text{size}(D)} \times -\log_2 \frac{\text{df}(P)}{N}$$

  - TF: number of occurrence of phrase P in a document / number of phrases in the document.
  - IDF: log of (number of documents containing P / number of documents)
  - First occurrence: number of phrases up to P / number of phrases in the document.
- Are calculated to be used in marking and extraction stages.

MARKING

- Candidate phrases are marked as yes if they are same with author assigned key phrases.

Discretization table

Feature	Discretization ranges				
	1	2	3	4	5
TF×IDF	< 0.0031	[0.0031, 0.0045)	[0.0045, 0.013)	[0.013, 0.033)	≥ 0.033
distance	< 0.0014	[0.0014, 0.017)	[0.017, 0.081)	≥ 0.081	

- According to marked or not TFxIDF and first ocurence values are grouped to be used in extraction stage (i.e. TFxIDF_not[0]++ for < 0.0031 value and not marked phrase).

EXTRACTION STAGE

- Test documents are loaded to the system (users can select how many to be loaded.
- The sentences / blocks are split into tokens.
- Same as training stage;
  - Inputs are cleaned.
  - Phrases are idendified.
  - Feature values are calculated.

PHRASE EXTRACTION

- Using training data values extraction probability is calculated ($P[TFxIDF\ |yes]$, $P[TFxIDF\ |no]$ ...).

$$P[TFxIDF\ |yes] = \frac{TFxIDF[yes]}{Y+N}$$

$$P[yes] = \frac{Y}{Y+N}\ x\ P[TFxIDF\ |yes]\ x\ P[distance\ |yes]$$

$$P[no] = \frac{N}{Y+N}\ x\ P[TFxIDF\ |no]\ x\ P[distance\ |no]$$

$$P[total] = \frac{P[yes]}{P[yes]+P[no]}$$

SENTENCE RANKING

- Using the extracting probabilities of the phrases, sentences are ranked and sorted in ascending order.
- At the end we may have similar sentences, so we eliminated these similar sentences as a final step.
- Sentences are displayed as a summary to the user according to threshold which is determined by the user.

SENTENCE RANKING

- With similarities eliminated

Otobüs ,Aras Nehri'ne uçtu :21 yaralı Bir yolcu otobüsünün Aras Nehri'ne uçması sonucu ilk belirlemelere göre 21 kişi yaralandı .
Son günlerde Aras nehri yakınlarında yaklaşık 5 bin kuşun öldüğü kaydedilen haberde ,''kuş gribinin İran'da görülme ihtimalinin günden güne arttığı ''ifade edildi .
Aras Nehri'ne uçan otobüsün şöförü gözaltında .
Kayır yolcuları arama çalışmaları sürüyor Kars'ın Sarıkamış ilçesinde Aras nehrine uçan yolcu otobüsünden yaralı kurtarılarak Erzurum'da tedavi altına alınan bir kişinin daha öldüğü ,ölü sayısının dokuza yükseldiği bildirildi .
Yaralı yolcular Horasan ve Sarıkamış Devlet Hastaneleri'ne kaldırıldı .

- Without eliminated

Otobüs ,Aras Nehri'ne uçtu :21 yaralı Bir yolcu otobüsünün Aras Nehri'ne uçması sonucu ilk belirlemelere göre 21 kişi yaralandı .
Öte yandan ,Kars Valisi Nevzat Turhan ,İstanbul'dan yola çıkmadan otobüste gerekli denetimlerin yapıldığını belirterek ,''Gerekli denetimler yapılıyor ama bu tür üzücü kazalar maalesef yaşanıyor .
Otobüs ,Aras Nehri'ne uçtu :21 yaralı .
Son günlerde Aras nehri yakınlarında yaklaşık 5 bin kuşun öldüğü kaydedilen haberde ,''kuş gribinin İran'da görülme ihtimalinin günden güne arttığı ''ifade edildi .
Yolcu otobüsü Aras Nehri'ne uçtu :3 ölü ,9 kayıp .

CONCLUSION

- Summarization method that uses key phrases is presented.
- Zemberek is used as stemmer.
- Feature values of the key phrases are used to rank sentences.
- Similar sentences are eliminated according to threshold.

QUESTIONS / ANSWERS