

# Summarization and QA

Ing. Roberto Tedesco, PhD

[roberto.tedesco@polimi.it](mailto:roberto.tedesco@polimi.it)



NLP – AA 20-21

# Introduction

- Summarization:  
*the process of distilling the most important information from a text to produce an abridged version for a particular task and user*
- Many typologies exist

# Different kinds of summaries

- Outlines
- Abstract
- Headlines (e.g., of a newspaper)
- Snippets (summarizing a Web pages on a search engine results page)
- Action Items (e.g., of spoken business meetings)
- Summaries (e.g., of emails)
- Compressed Sentences
- Answers to complex questions

# Classification: three axes

---

- **Single-document** versus **multiple-document** summarization
  - Single document → headline or outline
  - Multiple docs → condensation of the group
- **Generic summarization** versus **query-focused** summarization
  - Generic: no particular user or info is needed
  - Query-focused: summary is the answer to a query
- **Abstractive** versus **extractive** summarization
  - Most of systems are extractive (simpler to do)

# Extract vs abstract

Fourscore and seven years ago our fathers brought forth on this continent a new nation, conceived in liberty, and dedicated to the proposition that all men are created equal. Now we are engaged in a great civil war, testing whether that nation, or any nation so conceived and so dedicated, can long endure. We are met on a great battle-field of that war. We have come to dedicate a portion of that field as a final resting-place for those who here gave their lives that this nation might live. It is altogether fitting and proper that we should do this. But, in a larger sense, we cannot dedicate...we cannot consecrate...we cannot hallow... this ground. The brave men, living and dead, who struggled here, have consecrated it far above our poor power to add or detract. The world will little note nor long remember what we say here, but it can never forget what they did here. It is for us, the living, rather, to be dedicated here to the unfinished work which they who fought here have thus far so nobly advanced. It is rather for us to be here dedicated to the great task remaining before us...that from these honored dead we take increased devotion to that cause for which they gave the last full measure of devotion; that we here highly resolve that these dead shall not have died in vain; that this nation, under God, shall have a new birth of freedom; and that government of the people, by the people, for the people, shall not perish from the earth.


# Extract vs abstract

## **Extract from the Gettysburg Address:**

Four score and seven years ago our fathers brought forth upon this continent a new nation, conceived in liberty, and dedicated to the proposition that all men are created equal. Now we are engaged in a great civil war, testing whether that nation can long endure. We are met on a great battle-field of that war. We have come to dedicate a portion of that field. But the brave men, living and dead, who struggled here, have consecrated it far above our poor power to add or detract. From these honored dead we take increased devotion to that cause for which they gave the last full measure of devotion — that government of the people, by the people, for the people, shall not perish from the earth.

## **Abstract of the Gettysburg Address:**

This speech by Abraham Lincoln commemorates soldiers who laid down their lives in the Battle of Gettysburg. It reminds the troops that it is the future of freedom in America that they are fighting for.

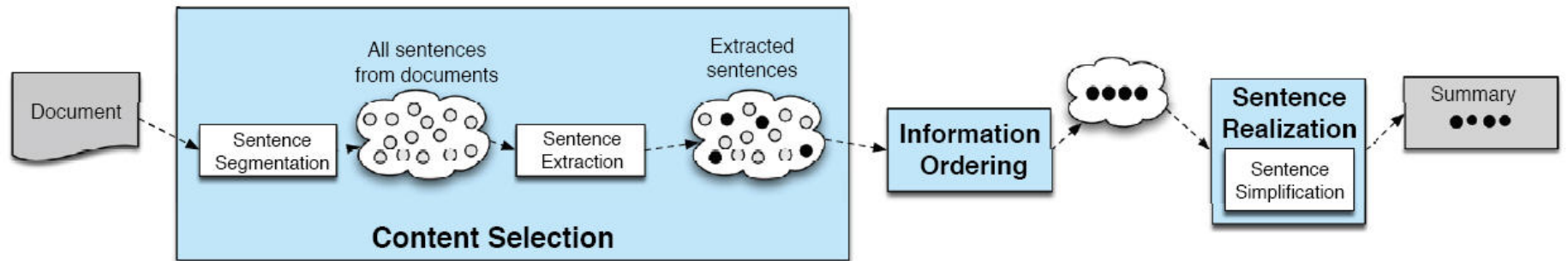


Generic summarization,  
Extractive summarization

# **SINGLE DOCUMENT**



# Pipeline of extractive summarization



1. **Content Selection:** choose pieces of text (*units*)
  - granularity (usually sentence or clause)
2. **Information Ordering:** choose the order of the extracted units
  - usually is easily solved: just keep the appearance order
3. **Sentence Realization:** clean up extracted units so they are fluent in their new context



# 1. Content selection

- Classification task: put each sentence into the “important” or “unimportant” classes
- Many methods:
  - A. Unsupervised Content Selection
  - B. Unsupervised Summarization based on Rhetorical Parsing
  - c. Supervised Content Selection

# A. Unsupervised Content Selection

- The content selection task is treated as a clusterization task:
  - compute salience (i.e., *informativeness*) of words
  - select sentences that have more salient words (i.e., define a threshold  $\Theta$  and keep sentences with “weight”  $> \Theta$ ; or define a size  $k$  and keep the  $k$  sentences with best “weight”)
- Many methods:
  - I. frequency-based  $\rightarrow$  TF-IDF
  - II. centroid-based  $\rightarrow$  log likelihood ratio
  - III. centrality-based methods

A collection of documents is needed

# Digression: The Vector Space Model (VSM)

- Documents  $d_j$  represented as *vectors of weights*  $\omega_{i,j}$
- The best choice for weights:
  - Normalized weights
  - Similarity: scalar product = cosine of angle between vectors
- Doc. collection represented by the  $M \times N$  matrix  $A = [\omega_{i,j}]$

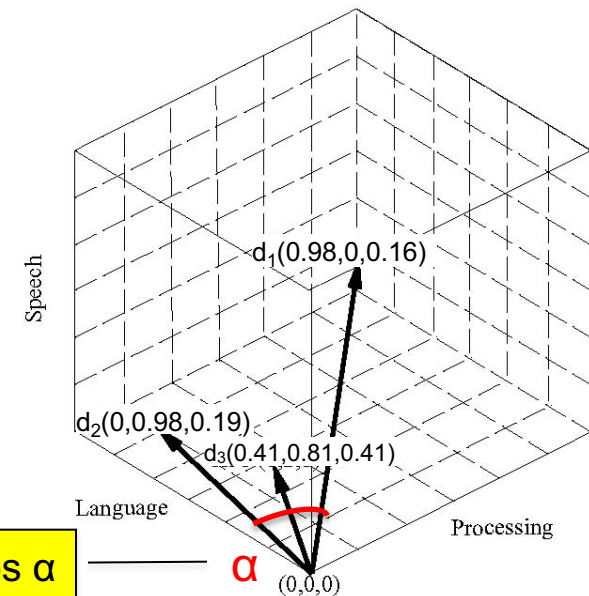
$$\vec{d}_j = (\omega_{1,j}, \omega_{2,j}, \omega_{3,j}, \dots, \omega_{M,j})$$

$$\vec{d}_k = (\omega_{1,k}, \omega_{2,k}, \omega_{3,k}, \dots, \omega_{M,k})$$

$$\text{sim}(\vec{d}_k, \vec{d}_j) = \sum_{i=1}^M \omega_{i,k} \cdot \omega_{i,j}$$

$$A = \begin{matrix} & \begin{matrix} \text{documents} \\ d_1 & d_2 & d_3 \end{matrix} \\ \begin{pmatrix} \omega_{1,1} & \omega_{1,2} & \omega_{1,3} \\ \omega_{2,1} & \omega_{2,2} & \omega_{2,3} \\ \omega_{3,1} & \omega_{3,2} & \omega_{3,3} \end{pmatrix} & \begin{matrix} t_1 \\ t_2 \\ t_3 \end{matrix} \end{matrix} \quad \begin{matrix} \\ \\ \text{terms} \end{matrix}$$

N: # of documents  
in collection  
M: # of unique terms  
(word types or  
word forms)



$$\text{sim}(q_1, q_2) = \cos \alpha$$

# Digression: the TF-IDF model

- *Term frequency* (the “density” measure):  
$$tf_{i,j} = (\# \text{ occurrences of the term } t_i \text{ in doc. } d_j) / (\# \text{ terms in doc. } d_j)$$
- *Inverse document frequency* (the “rarity” measure):  
$$idf_i = \log(N / (\# \text{ docs containing } t_i))$$
- *Weights* are defined as:  
$$\omega_{i,j} = tf_{i,j} \cdot idf_i$$
- If a term  $t_i$  is “dense” in a given document  $d_j$ , but rare in the collection, it is highly relevant for  $d_j$ 
  - It represents  $d_j$
- A *term*, could be a word type or a word form

# I. Frequency based

- The TF-IDF: high weight to words that appear frequently in the current document, but rarely in the overall document collection
- For each word type  $w_i$  in the document  $j$ :  
 $weight_j(w_i) = \text{tf}_{i,j} \cdot \text{idf}_i$
- Then, the weight of a sentence  $s_k$  in the document  $j$  is the average weight of its non-stop words:

$$weight_j(s_k) = \sum_{w_i \in \text{Non\_stop}(s_k)} \frac{weight_j(w_i)}{|\text{Non\_stop}(s_k)|}$$

## II. centroid based

- Find *signature* word types
  - These words “characterize” the document
  - Sentences are weighted according to the presence of such words
- The log likelihood ratio  $D = -2\log(\lambda(w_i))$  is one of such methods

# The log likelihood ratio

- $D = -2\log(\lambda(w_i))$ : how important a word type  $w_i$  is for the current document with respect to the document collection
  - $D$ : discrepancy of the *observed* word frequencies in the current document from the values which we would expect to see *if the word frequencies were the same in the current document and in the collection*
- Large discrepancy  $\rightarrow$  large value of  $D \rightarrow$  the difference between the word frequencies in the current document and in the collection is statistically significant



# The log likelihood

- Expected value of  $C_{doc}(w_i)$  if equal frequency:

$$E_{doc}[w_i] = \frac{N_{doc}}{N_{doc} + N_{oth}} \cdot (C_{doc}(w_i) + C_{oth}(w_i))$$

- Expected value of  $C_{oth}(w_i)$  if equal frequency:

$$E_{oth}[w_i] = \frac{N_{oth}}{N_{doc} + N_{oth}} \cdot (C_{doc}(w_i) + C_{oth}(w_i))$$

- And the log-likelihood ratio is:

$$D = -2 \cdot \log(\lambda(w_i)) = 2 \cdot \left[ C_{doc}(w_i) \cdot \log\left(\frac{C_{doc}(w_i)}{E_{doc}(w_i)}\right) + C_{oth}(w_i) \cdot \log\left(\frac{C_{oth}(w_i)}{E_{oth}(w_i)}\right) \right]$$

# The log likelihood

a chi-squared distribution with  $k=1$  degrees of freedom

- For large corpora:  $D \sim \chi^2(k=1)$
- Thus, if  $D > 10.8$ ,  $w_i$  is significant for the current document, with at least 0.99 of significance level
- For each word type  $w_i$  in the document  $j$ :

$$weight_j(w_i) = \begin{cases} 1; & -2 \cdot \log(\lambda(w_i)) > 10.8 \\ 0; & otherwise \end{cases}$$

- And the weight of a sentence  $s_k$  is:

$$weight_j(s_k) = \sum_{w_i \in \text{Non\_stop}(s_k)} \frac{weight_j(w_i)}{|\text{Non\_stop}(s_k)|}$$

### III. centrality based

- Centrality-based methods compute distances between each candidate sentence and each other sentence, of the current document
  - Choose sentences that are on average closer to other sentences
- To compute centrality:
  - Represent sentences as a bag-of-words vector
  - Each sentence  $s$  of the document is then assigned a centrality score:

$$centrality(s) = \frac{1}{K} \sum_y \text{tf\_idf\_cosine}(s, y)$$

## B. Unsupervised Summarization Based on Rhetorical Parsing

- Use of Coherence Relations (such as Rhetorical Structure Theory)
- First apply of a Discourse Parsing to compute the coherence relation graph or parse tree
  - Nuclear units are the important parts to put into the summary
- More on this in the course section about “discourse”...

## B. Unsupervised Summarization Based on Rhetorical Parsing

coherence  
relation

JUSTIFICATION

With its distant orbit - 50 percent farther from  
the sun than Earth - and slim atmospheric blanket,

Mars experiences frigid weather conditions

satellite

nucleus

## C. Supervised Content Selection

- Weighting words is only a single cue for finding extractworthy sentences
- Many other cues exist:
  - position of the sentence: sentences at the very beginning or end of the document tend to be more important
  - the length of each sentence
  - and so on...
- We'd like a method that can weigh and combine all these cues

# What we need

- Supervised model
  - each sentence is a sample
  - goal: classify each sample as extractworthy or not
  - we need a training set of documents paired with human-created summary extracts
- The corpus
  - Since summaries are *extracts*, each sentence in the summary is taken from the document
  - we assign a label to every sentence in the document: 1 if it appears in the extract, 0 if it doesn't



# What we need

- A set of features is computed for the sentence to classify
  - see next slide for an example
- Then, choose a classification model

# Features

<b>position</b>	<p>The position of the sentence in the document. For example, Hovy and Lin (1999) found that the single most extract-worthy sentence in most newspaper articles is the title sentence. In the Ziff-Davis corpus they examined, the next most informative was the first sentence of paragraph 2 (P1S1), followed by the first sentence of paragraph 3 (P3S1); thus the list of ordinal sentence positions starting from the most informative was: T1, P2S1, P3S1, P4S1, P1S1, P2S2,...</p> <p>Position, like almost all summarization features, is heavily genre dependent. In <i>Wall Street Journal</i> articles, they found the most important information appeared in the following sentences: T1, P1S1, P1S2,...</p>
<b>cue phrases</b>	<p>Sentences containing phrases like <i>in summary</i>, <i>in conclusion</i>, or <i>this paper</i> are more likely to be extract worthy. These cue phrases are very dependent on the genre. For example, in British House of Lords legal summaries, the phrase <i>it seems to me that</i> is a useful cue phrase (Hachey and Grover, 2005).</p>
<b>word informativeness</b>	<p>Sentences that contain more terms from the <b>topic signature</b>, as described in the previous section, are more extract worthy.</p>
<b>sentence length</b>	<p>Very short sentences are rarely appropriate for extracting. We usually capture this fact by using a binary feature based on a cutoff (true if the sentence has more than, say, five words).</p>
<b>cohesion</b>	<p>Recall from Chapter 21 that a <b>lexical chain</b> is a series of related words that occurs throughout a discourse. Sentences that contain more terms from a lexical chain are often extract worthy because they are indicative of a continuing topic (Barzilay and Elhadad, 1997). This kind of cohesion can also be computed by graph-based methods (Mani and Bloedorn, 1999). The PageRank graph-based measures of sentence centrality discussed above can also be viewed as a coherence metric (Erkan and Radev, 2004).</p>

# Generalizing the corpus

- When we write summaries, we very often use phrases and sentences from the document to compose the summary
  - even when writing **abstractive** summaries!
- But we don't use only extracted sentences:
  - combine two sentences into one
  - change some of the words in the sentences
  - write completely new abstractive sentences
- The assumption that all the sentences into the corpus of summaries come from the original document is restrictive

# Alignment

- Thus, assume that summaries in the corpus are *not* extracts
  - it does not hold that each sentence in the summary is taken from the document
  - we need to align each document's sentence with its summary sentence(s)
- Possible algorithms for alignment:
  - align document and abstract sentences with the longest common subsequences of non-stopwords
  - edit distance
  - WordNet-based distance

### 3. Sentence Realization: sentence simplification

- Rules to select parts of the sentence to prune or keep; often by running a parser/chunker over the sentences
  - For example: some rules for pruning:

**appositives**

Rajam, ~~28, an artist who was living at the time in Philadelphia,~~  
found the inspiration in the back of city magazines.

**attribution clauses**

Rebels agreed to talk with government officials, ~~international  
observers said Tuesday.~~

**PPs without  
named entities**

The commercial fishing restrictions in Washington will not be  
lifted [SBAR unless the salmon population 329 increases [PP ~~to  
a sustainable number~~]

**initial adverbials**

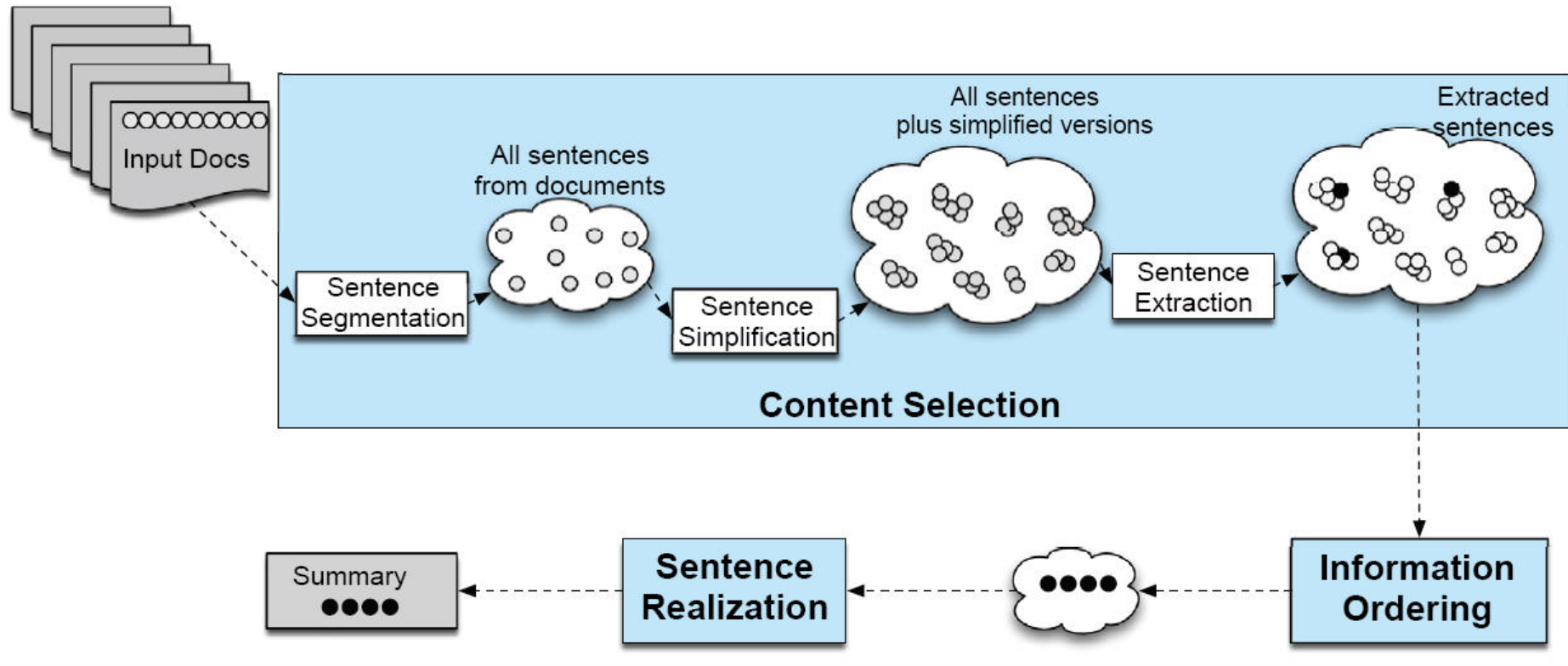
“For example”, “On the other hand”, “As a matter of fact”, “At  
this point”



Generic summarization,  
Extractive summarization

# MULTIPLE DOCUMENTS

# Pipeline of extractive summarization for Multi-Document Summarization



1. **Content Selection**
2. **Information Ordering**
3. **Sentence Realization**



# 1. Content Selection in Multi-Document Summarization

- Multiple document summarization  
→ great amount of redundancy
- The summary should not contain redundancy
  - Algorithms for multi-document summarization focus on ways to avoid redundancy
  - Adding a new sentence to the summary, be sure the sentence doesn't overlap too much with sentences already in the summary
  - Redundancy factor: similarity between a candidate sentence and the sentences into the summary
- A sentence is penalized if it is too similar

# How to select

- Maximal Marginal Relevance (MMR)
  - penalization term for redundant sentence:
$$\text{MMR\_penalization\_factor}(s) = \lambda \cdot \max_{s_i \in \text{Summary}} \text{Sim}(s, s_i)$$
  - Used to lowers weight of redundant sentences
- Clustering:
  - apply a clustering algorithm to all the sentences in the documents to be summarized
  - produce a number of clusters of related sentences
  - select a single (centroid) sentence from each cluster into the summary

## 2. Information Ordering in Multi-Document Summarization

- To decide how to concatenate the extracted sentences into a coherent order
- Many methods:
  - A. Chronological ordering
  - B. Coherence
  - C. Centering

## A. Chronological ordering

- For sentences extracted from news stories
- Use the dates associated with the story
- It turns out that pure chronological ordering can produce summaries which lack cohesion
  - this problem can be addressed by ordering slightly larger chunks of sentences

## B. Coherence

- Coreference; a coherence discourse is one in which entities are mentioned in *coherent patterns*
- An ordering heuristic: *lexical cohesion*
  - use the standard TF-IDF cosine distance between each pair of sentences
  - choose the overall ordering that minimizes the average distance between neighboring sentences

## C. Centering

- Each discourse segment has a salient entity: the *focus*
- Coherent discourse if:
  - the focus appears as certain *syntactic realizations* (i.e. as subject or object)
  - the focus appears in certain *transitions* between these realizations (e.g., if the same entity is the subject of adjacent sentences)
- We prefer orderings in which the transition between entity mentions is a preferred one

# Entity grid

corpus

entity grid

- 1 [The Justice Department]<sub>S</sub> is conducting an [anti-trust trial]<sub>O</sub> against [Microsoft Corp.]<sub>X</sub>
- 2 [Microsoft]<sub>O</sub> is accused of trying to forcefully buy into [markets]<sub>X</sub> where  
[its own products]<sub>S</sub> are not competitive enough to unseat [established brands]<sub>O</sub>
- 3 [The case]<sub>S</sub> resolves around [evidence]<sub>O</sub> of [Microsoft]<sub>S</sub> aggressively  
pressuring [Netscape]<sub>O</sub> into merging [browser software]<sub>O</sub>
- 4 [Microsoft]<sub>S</sub> claims [its tactics]<sub>S</sub> are commonplace and good economically.

	Department	Trial	Microsoft	Markets	Products	Brands	Case	Netscape	Software	Tactics
1	S	O	X	-	-	-	-	-	-	-
2	-	-	O	X	S	O	-	-	-	-
3	-	-	S	O	-	-	S	O	O	-
4	-	-	S	-	-	-	-	-	-	O

- We can learn preferred entity positions
- For example: the transitions {X,O,S,S} for the entity Microsoft
  - “Microsoft” in a discourse is introduced first in oblique or object position and then only later appears in subject position
- Then, select sentence ordering that respects preferred entity positions



# Selection and ordering together: HMM

- HMM: selection and ordering together
  - clusterize sentences
  - each cluster  $c$  is an (unnamed) *topic*: hidden state
  - observations correspond to sentences  $s$
  - HMM transition probability distribution:
$$P(c_j \mid c_i) = C_{\text{collection}}(\text{sent}(c_i) \rightarrow \text{sent}(c_j)) / C_{\text{collection}} \text{sent}(c_i)$$
  - HMM emission probability distribution:
$$P(s \mid c) = C_{\text{collection}}(s \in \text{sent}(c)) / \sum_{s'} C_{\text{collection}}(s' \in \text{sent}(c))$$
- $P(c_j \mid c_i)$  implicitly represents information-ordering facts:
  - $P(c_j \mid c_i)$  is high if often sentences in  $c_i$  precedes sentences in  $c_j$ ; in other words:  $c_i$  is “before”  $c_j$
  - select the ordering, among all the candidates, that the HMM assigns the highest probability to
- $P(s \mid c)$  implicitly represents a way for selecting a “reference” sentence for a cluster
  - per each cluster, select the sentence with  $\max P(s \mid c)$

# 3. Sentence Realization in Multi-Document Summarization

## Original summary:

Presidential advisers do not blame **O'Neill**, but they've long recognized that a shakeup of the economic team would help indicate **Bush** was doing everything he could to improve matters. **U.S. President George W. Bush** pushed out **Treasury Secretary Paul O'Neill** and top economic adviser Lawrence Lindsey on Friday, launching the first shakeup of his administration to tackle the ailing economy before the 2004 election campaign.

## Rewritten summary:

Presidential advisers do not blame **Treasury Secretary Paul O'Neill**, but they've long recognized that a shakeup of the economic team would help indicate **U.S. President George W. Bush** was doing everything he could to improve matters. **Bush** pushed out **O'Neill** and White House economic adviser Lawrence Lindsey on Friday, launching the first shakeup of his administration to tackle the ailing economy before the 2004 election campaign.

- Incoherence; e.g.,: the full name “U.S. President George W. Bush” occurs only after the shortened form “Bush”

## How to clean

- The ordering chosen for the extracted sentences may not respect coherence rules (see example before)
- A coreference resolution algorithm must be applied to the output, extracting names and applying cleanup rewrite rules; e.g.:
  - use the full name at the first mention, and just the last name at subsequent mentions.
  - use a modified form for the first mention, but remove appositives or premodifiers from any subsequent mentions
  - ...



# **FOCUSED SUMMARIZATION**

# Focused Summarization & QA

- Summarization techniques are often used to build answers to complex questions
- Methods:
  - A. Slightly modify the algorithms for multiple-document summarization
    - to make use of the query
    - i.e., extracting sentences containing at least one word overlapping with the query
  - B. Use Information Extraction methods
- We'll see method B

# Focused Summarization & QA: Information Extraction

Possible query pattern

<b>Definition</b> <b>What's</b> (X where X is not in Person)	
<b>genus</b>	The Hajj is a type of ritual
<b>species</b>	the annual hajj begins in the twelfth month of the Islamic year
<b>synonym</b>	The Hajj, or Pilgrimage to Mecca, is the central duty of Islam
<b>subtype</b>	Qiran, Tamattu', and Ifrad are three different types of Hajj
<b>Biography</b> <b>Who's</b> (X where X is in FamousPerson)	
<b>dates</b>	was assassinated on April 4, 1968
<b>nationality</b>	was born in Atlanta, Georgia
<b>education</b>	entered Boston University as a doctoral student
<b>Drug efficacy</b> <b>How good is</b> (X where X is in Drug)	
<b>population</b>	37 otherwise healthy children aged 2 to 12 years
<b>problem</b>	acute, intercurrent, febrile illness
<b>intervention</b>	acetaminophen (10 mg/kg)
<b>outcome</b>	ibuprofen provided greater temperature decrement and longer duration of antipyresis than acetaminophen when the two drugs were administered in approximately equal doses

- Handled query *classes*: **Definition**, **Biography**, ...
- For each query class, multiple patterns are defined
  - Usually, by guided examples
- Patterns define the keyword(s) to search for with the IR engine

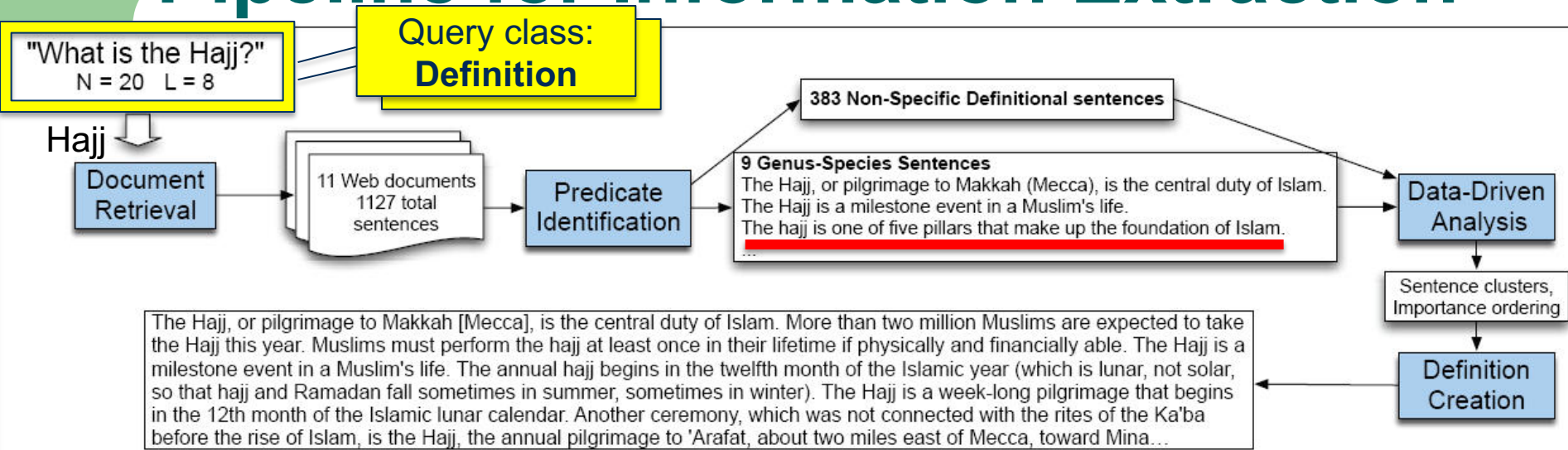
# Focused Summarization & QA: Information Extraction

	Definition	Possible result pattern
<b>genus</b> <b>species</b> <b>synonym</b> <b>subtype</b>	<p>The <u>Hajj</u> is a <u>type of ritual</u></p> <p>the annual hajj begins in the twelfth month of the Islamic year</p> <p>The Hajj, or Pilgrimage to Mecca, is the central duty of Islam</p> <p>Qiran, Tamattu', and Ifrad are three different types of Hajj</p>	<p>The x is GEN</p>
	Biography	
<b>dates</b> <b>nationality</b> <b>education</b>	<p>was assassinated on April 4, 1968</p> <p>was born in Atlanta, Georgia</p> <p>entered Boston University as a doctoral student</p>	
	Drug efficacy	
<b>population</b> <b>problem</b> <b>intervention</b> <b>outcome</b>	<p>37 otherwise healthy children aged 2 to 12 years</p> <p>acute, intercurrent, febrile illness</p> <p>acetaminophen (10 mg/kg)</p> <p>ibuprofen provided greater temperature decrement and longer duration of antipyresis than acetaminophen when the two drugs were administered in approximately equal doses</p>	

- *Info extraction typologies* are the expected results: **genus**, ...
- For each info extraction typology, multiple patterns are defined
  - Usually, by guided examples
- Patterns define the important info to extract from the IR results



# Focused Summarization & QA: Pipeline for Information Extraction



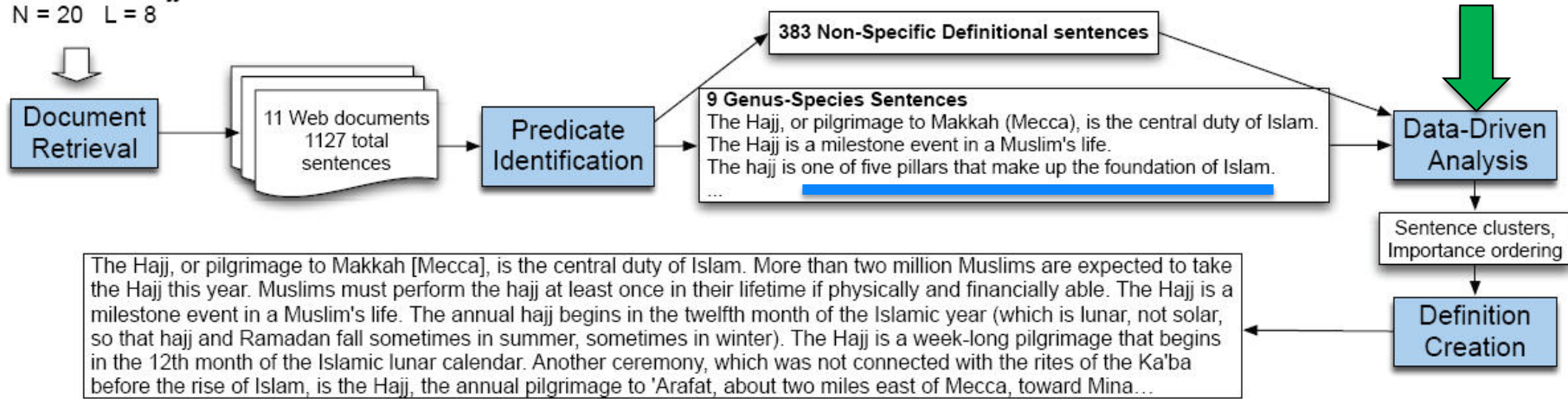
- Patterns to recognize the query class and extract the term to be defined from the question (e.g., x="Hajj")
- Queries are sent to the IR (e.g., "Hajj") → retrieved documents
- Patterns on results, to search for info extraction typologies
  - I know the query class (e.g.: **Definition**) → I know the info extraction typologies to search for into the results (e.g., **genus**)
  - Label each sentence with its "info extraction typology":
    - 383 "Non-specific definitional" sentences
    - 9 "Genus-Species" sentences (e.g.: "The hajj is one of five...")



# Focused Summarization & QA: Pipeline for Information Extraction

"What is the Hajj?"

N = 20 L = 8



- Extract info from sentences

- ➡ – Patterns defined for a given info extraction typology permits to extract useful info to add to the answer (e.g.: GEN="one of five...")

- Apply class-specific templates for answer generation

- ➡ – E.g.: For questions of class **Definition** we might define:

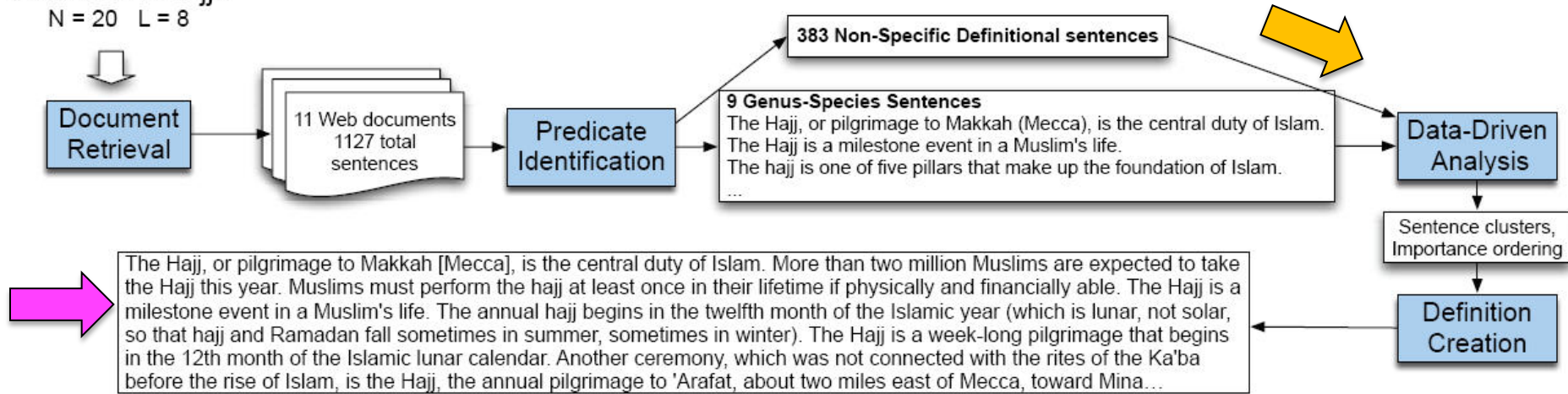
**"The X, or SYNONYM, is GEN. SUBTYPE. TIME ..."**

- Variables extracted by patterns defined for the info extraction typologies

# Focused Summarization & QA: Pipeline for Information Extraction

"What is the Hajj?"

N = 20 L = 8



- Add to the answer additional sentences that might not fall into a specific info extraction typology
  - Sort of “context info”
  - e.g., the 383 sentences tagged with “Non-specific definitional”
- As in multiple document summarization: sentence clustering, importance ordering, ...



# SUMMARIZATION EVALUATION

# Evaluation

- Extrinsic (task based): put the system in action
- Intrinsic (task independent)
- Two examples of intrinsic methodologies:
  - ROUGE-N
  - Pyramid

# Recall-Oriented Evaluation (ROUGE-N)

- For a given document  $d$ , measures the amount of  $N$ -gram overlap between the candidate summary  $S_c$  and human-generated Reference Summaries for  $d$
- ROUGE-2 is a measure of the bigram *recall*

# bigrams in common between  
*ReferenceSummaries* and  $S_c$

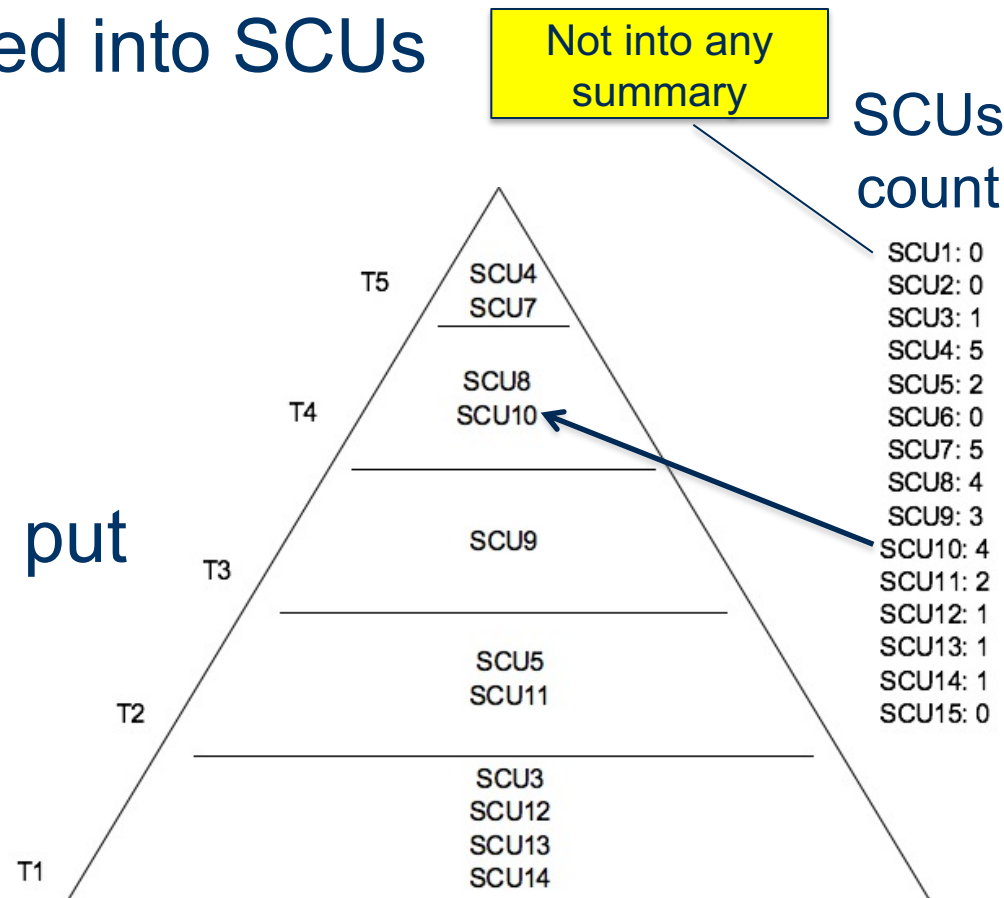
$$ROUGE2(S_c) = \frac{\sum_{S \in \text{ReferenceSummaries}} \sum_{\text{bigram} \in S} C(\text{bigram}, S_c)}{\sum_{S \in \text{ReferenceSummaries}} \sum_{\text{bigram} \in S} C(\text{bigram}, S)}$$

# bigrams of *ReferenceSummaries*

- For each *unique bigram* in each summary  $S$ :  
 $C(\text{bigram}, S_c)$  = count of *bigram* in  $S_c$   
 $C(\text{bigram}, S)$  = count of *bigram* in  $S$

# Pyramid

- SCU (Summary Content Unit): The smallest text unit carrying a specific information (e.g. sentence)
- The text is subdivided into SCUs
  - e.g., 15
- $n$  humans generate summaries
  - e.g.,  $n=5$
- Level  $T_n$ :  $n$  humans put those SCUs in their summaries
  - e.g., SCU10 is in  $T_4$



# Pyramid

$$\text{Recall} = \frac{\sum_{i=1}^n D_i}{\sum_{i=1}^n T_i}$$

$$\text{Precision} = \frac{\sum_{i=1}^n D_i}{\sum_{i=0}^n D_i}$$

- $D_i$ : #SCUs found into the generated summary and that are at the  $i$ -th level
- $T_i$ : #SCUs at the  $i$ -th level
- $D_0$ : #SCUs found into the generated summary and that are NOT in the pyramid

$$\text{Accuracy} = \frac{\text{truePos} + \text{trueNeg}}{\text{truePos} + \text{trueNeg} + \text{falsePos} + \text{falseNeg}}$$

- Accuracy: #SCUs correctly classified (inside or outside the pyramid) divided by the total number of #SCUs