USTHB university Algiers, Algeria
Electronics and Computer engineering Faculty (FEI)
Signal Processing Laboratory

Topic Identification of Noisy Arabic Texts Using Graph Approaches

Key Words :

Mr K. ABAINIA Dr S. OUAMOUR Prof H. SAYOUD

Talk Outline

1. Background
2. Corpus
3. Preprocessing
4. Topic Identification based Graph Approaches
5. Experimental results
6. Summary

What is Topic Identification ?

Topic Identification is the task of automatically recognizing the **subject** or the **theme** in which the text is written.

Automatic text categorization by attributing one or more labels from a **predefined** set of topics.

Applications of Topic Categorization

- **Newswires:** news are organized and archived by subject categories *(e.g. Politics, Economy, Sports, etc.);*
- **Academic articles:** papers are classified by domains and sub-areas;
- **Emails routing:** directing received emails to a specific mailbox depending on the topic;
- **Civil security:** predicting manifestations and/or terrorists' plots by automatically analyzing on-line conversations;

Motivation

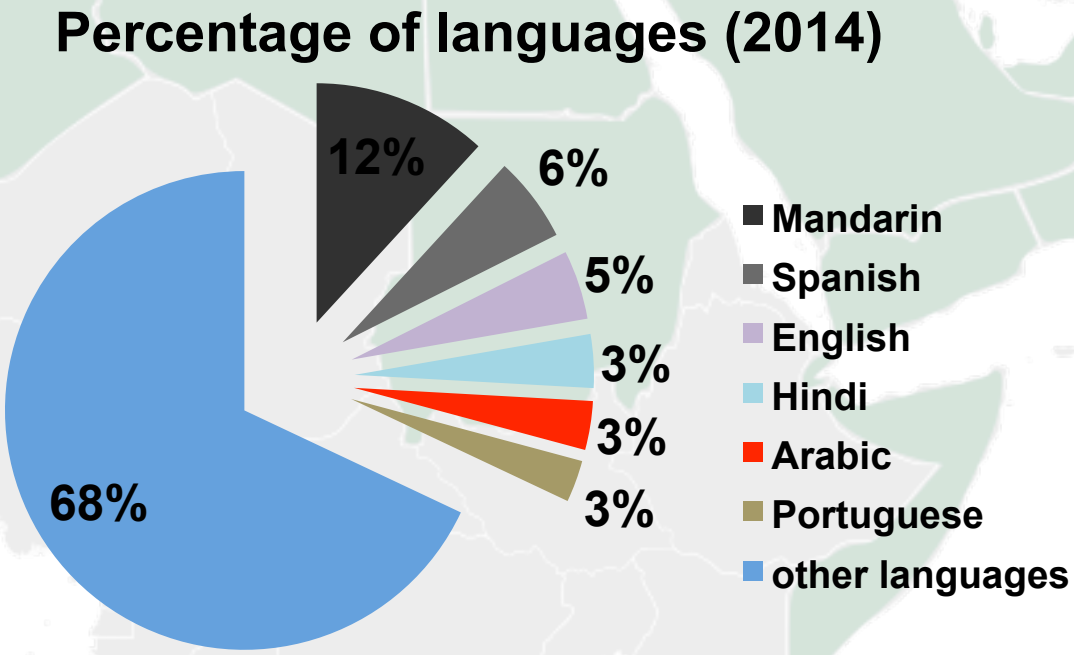
❖Many Topic Identification researches have been evaluated on **long** and **well written** texts (e.g. Scientific papers and Newspaper articles).

❖Many researches have been undergone on European languages and Asian languages, except the Arabic language (**few works**).

❖**Arabic language** is the more difficult one having a **complex** morphology and a **large** vocabulary.

Statistics (1)

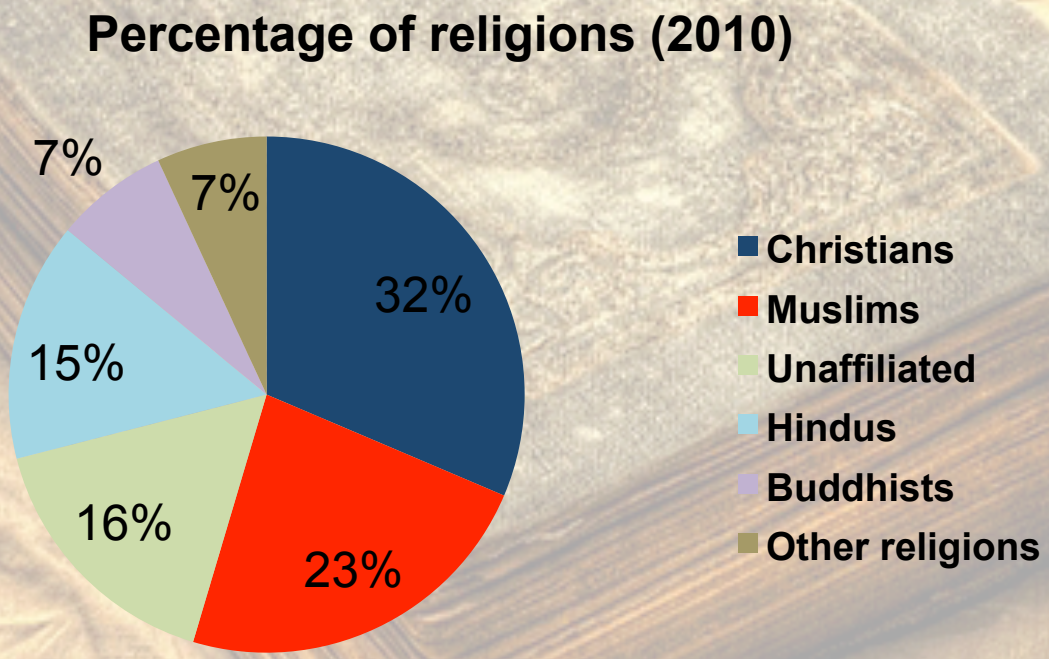
Arabic was the 5th most widely-spoken language, and is the tongue language of 422 million people in 22 countries.



https://www.cia.gov/library/publications/the-world-factbook/

Statistics (2)

As the religious language of the Quran (7th century), it was rapidly expanded during the rise of Islam in the 8th century



PEW Research Center

Arabic language characteristics

❖Alphabet set consists of **28 main letters** with other forms taken by some characters (e.g. *Alif* "ألف", *Yaa* "ياء" and *Taa* "تاء").

❖There is **no capitalization** in Arabic (i.e. capital and small letters)

❖Letter can have different shapes depending on its location in the word.

separated	end	middle	beginning
ي	ـي	ـيـ	يـ

❖Word **meaning** is often determined by **diacritics** (or vowels).

كِلَامْ	كَلَامْ
(wound)	(speech)

❖Letter repetition twice is replaced by ***Shadda*** character "ّ"

❖Some conjunctions like "و" (*AND*) are welded to the following word, which makes the preprocessing quite difficult.

Talk Outline

1. Background
2. Corpus
3. Preprocessing
4. Topic Identification based Graph Approaches
5. Experimental results
6. Summary

ANTSIX Corpus

Text length ranges between 32 and 318 words

Difficulty

Main difficulty = Noisy texts (discussion forum texts)

- **Citations in other languages**
- **URLs**
- **Typing errors**
- **Tags** (e.g. hash tags, user tags...)
- **Insignificant characters** (e.g. emoticons)
- **Abbreviations**
- **Letters mistakenness** (e.g. the letter "ظ" and the letter "ض")

Talk Outline

1. Background
2. Corpus
3. **Preprocessing**
4. Topic Identification based Graph Approaches
5. Experimental results
6. Summary

Text preprocessing (step 1)

- Read the text as UTF-8 text.
- Strip some characters:
  - insignificant characters.
  - French and English characters.
  - Arabic diacritics.
- Separate contracted words (i.e. Replace "/" and "-" with white spaces).
- Strip multiple word delimiters (i.e. white space, "\n" and "\r").
- Normalize some letters:
  - ✓Replace *Alif* with different forms ("إ" ,"أ" and "آ") by *Alif* bare ("ا").
  - ✓Replace *Alif MaqSura* ("ى") by *Yaa* ("ي").

Text preprocessing (step 2)

- Extract a list of words.
- Remove stop words (**600 stop words**).
- Stem the rest of words (remove prefixes and suffixes).

Talk Outline

1. Background
2. Corpus
3. Preprocessing
4. **Topic Identification based Graph Approaches**
5. Experimental results
6. Summary

Approaches of topic identification

- Three graph approaches
  - **LIGA**
  - **TIGA1**
  - **TIGA2**
- **Nodes** represent the **word weights** and **edges** represent **word successions**
- The graph is represented by the following quintuple $G_i = (V_i, E_i, \mathcal{L}_i, W_{vi}, W_{ei})$
  - V_i and E_i are respectively a set of nodes and a set of edges.
  - $\mathcal{L}_i : V_i \rightarrow T$ Function used to assign vertices to the graph.
  - $W_{vi} : V_i \times T \rightarrow \mathbb{N}$ Function to assign weights to vertices.
  - $W_{ei} : E_i \times T \rightarrow \mathbb{N}$ Function to assign weights to edges.
- **Resultant graphs can be easily interpreted by human (visual analytics)**

LIGA approach (*training*)

Training doc $\in t_i$ → A list of words *W*

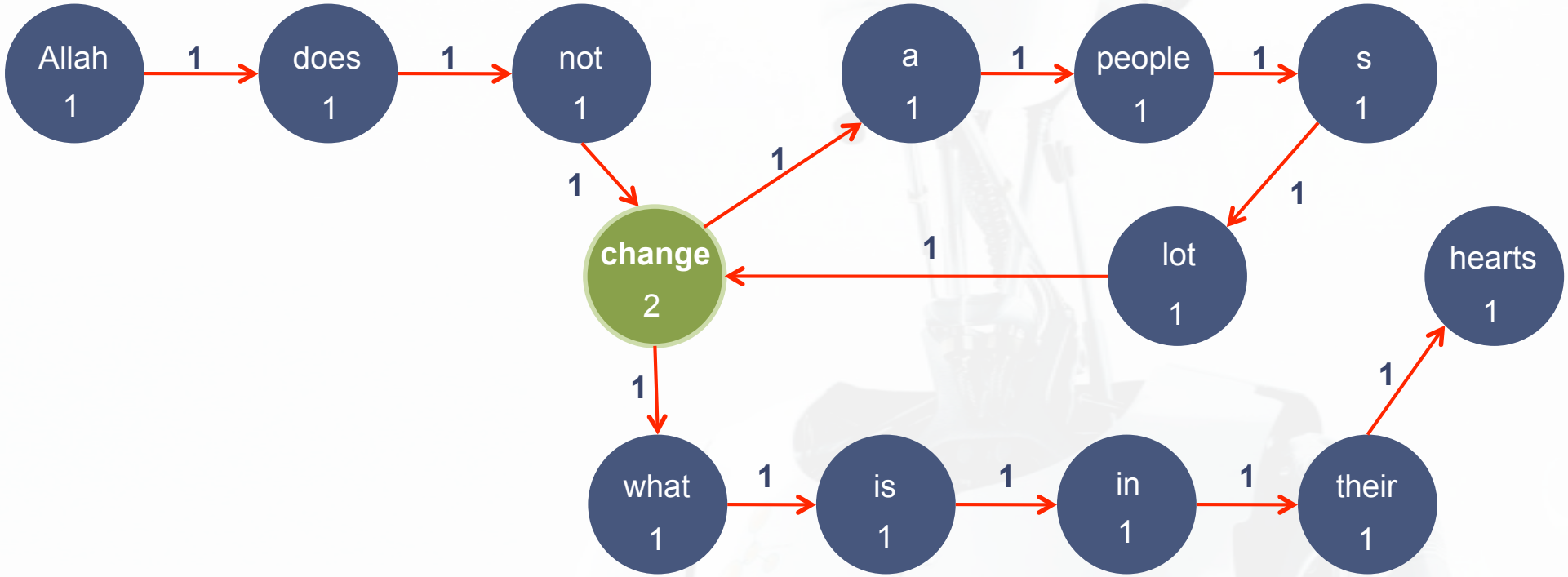
$$\forall w \in W \rightarrow \begin{cases} W_{vi}(v,t_i) = W_{vi}(v,t_i) + 1 & \text{if } v \in V_i \\ \mathcal{L}_i(v) = w \wedge W_{vi}(v,t_i) = 1 & \text{otherwise} \end{cases}$$

$$\forall w_j, w_{j+1} \in W \rightarrow \begin{cases} W_{ei}(e,t_i) = W_{ei}(e,t_i) + 1 & \text{if } e \in E_i \\ \text{edge } e \text{ is created} \wedge W_{ei}(e,t_i) = 1 & \text{otherwise} \end{cases}$$

Training example

"Allah does not change a people´s lot unless they change what is in their hearts"

[Allah] [does] [not] [**change**] [a] [people] [s] [lot] [unless] [they] [**change**] [what] [is] [in] [their] [hearts]



LIGA approach (*classification*)

Unlabeled text → path of words π (***consecutive words***)

Similarity = path matching function

$$\forall t_i \rightarrow PM(t_i) = 0$$

$$\forall w \in \pi \rightarrow PM(t_i) = \begin{cases} PM(t_i) + W_{vi}(v, t_i) & if\ v \in G_i \\ PM(t_i) & else \end{cases}$$

$$\forall w_j, w_{j+1} \in \pi \rightarrow PM(t_i) = \begin{cases} PM(t_i) + W_{ei}(e, t_i) & if\ e \in G_i \\ PM(t_i) & else \end{cases}$$

$$topic = argmax_{t_i \in T}(PM(t_i))$$

TIGA1

TIGA1 → Biasing the **LIGA** **node** weights using **TF-IDF** method.

$$tfidf(v,t_i) = W_{vi}(v,t_i) * idf_v$$

$W_{vi}(v,t_i)$ is the weight of the node v in the graph

idf_v is the inverse graph frequency

$$idf_v = \log(^{n}/_{M_v})$$

TIGA2

TIGA2 → Biasing the **TIGA1** **edge** weights using **TF-IDF** method.

$$tfidf(e,t_i) = W_{ei}(e,t_i) * idf_e$$

$W_{ei}(e,t_i)$ is the weight of the edge e in the graph

idf_e is the inverse graph frequency

$$idf_e = \log(n / M_e)$$

Talk Outline

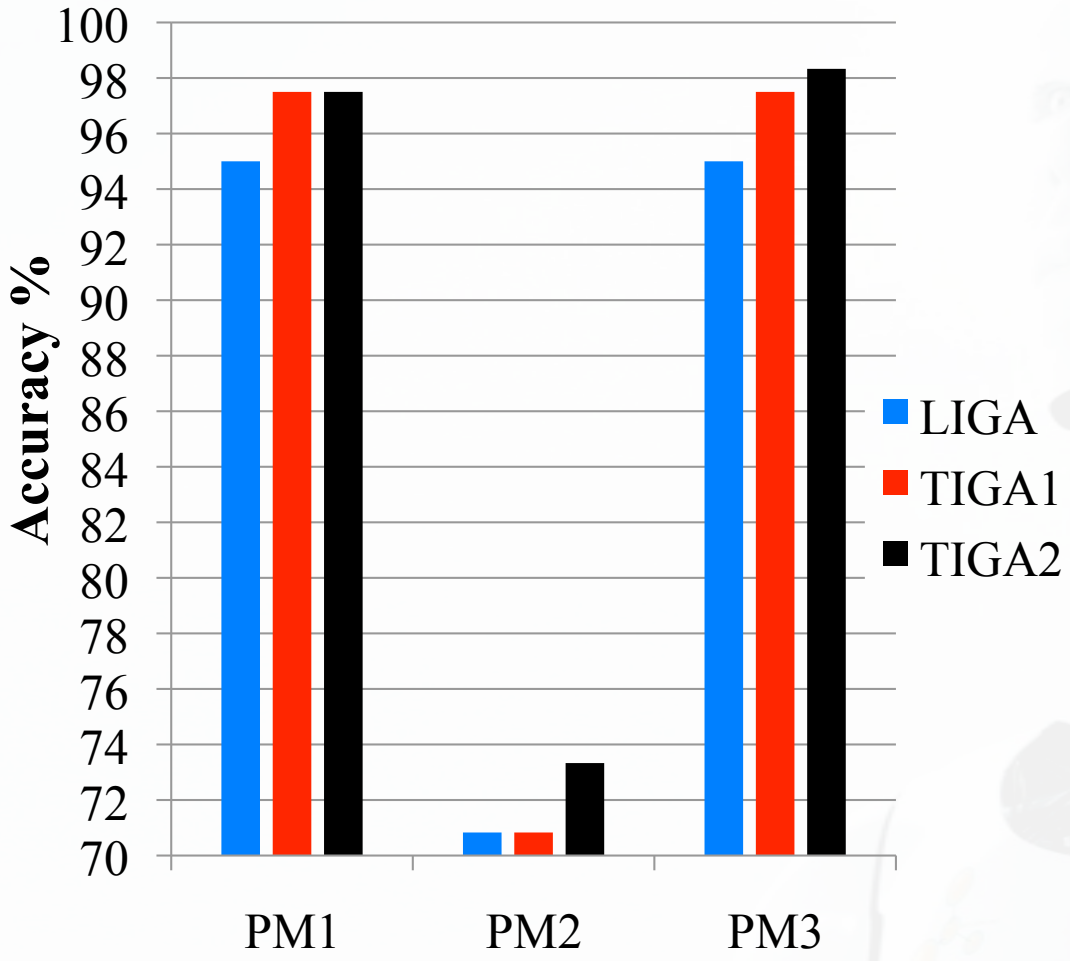
1. Background
2. Preprocessing
3. Corpus
4. Topic Identification based Graph Approaches
5. **Experimental results**
6. Summary

Experiment setup

- ANTSIX corpus:
  - 60% was used in the training.
  - 40% was reserved for the test.

- **Three** path matching functions are used:
  - PM1: uses only node weights.
  - PM2: uses only edge weights.
  - PM3: uses node and edge weights both.

Results (accuracies)



- **LIGA** is **worse** than **TIGA1** and **TIGA2**.
- **TIGA2** **is more accurate** than the two others (achieved the best accuracy).
- **PM2** produced **worse** performances comparing to **PM1** and **PM3**.
- **LIGA** and **TIGA1** can be **optimized** by using **PM1** instead of the original **PM3**.

Talk Outline

1. Background
2. Preprocessing
3. Corpus
4. Topic Identification based Graph Approaches
5. Experimental results
6. **Summary**

Conclusion

- Several experiments of **topic identification** were conducted on **noisy Arabic forum texts**.

- **Contributions**
  - ❖ **ANTSIX corpus**
    - ✓300 **noisy** Arabic texts
    - ✓6 different topics
  - ❖ **Preprocessing**
    - ✓Characters removing and normalizing
    - ✓Stop words removing(600 stop words)
    - ✓Words stemming
  - ❖ **Three approaches**
    - ✓LIGA
    - ✓Two improvements (i.e. TIGA1 and TIGA2)

- **Comparison** between **3 PM** functions
  - ✓TIGA1 and TIGA2 **outperformed** LIGA
  - ✓TIGA2 is the **more accurate**
  - ✓Original PM (PM3) can be **optimized** as PM1
  - ✓PM2 is not suitable in noisy Arabic texts

Perspective

- Benchmark the proposed approaches
  - ❖ **Evaluate TIGA1 and TIGA2 on a large corpus**
  - ❖ **Compare the two algorithms with other well known tools**
- Test other techniques of weighting instead TF-IDF

Thank you

Kheireddine Abainia
abainia@hotmail.fr

Siham Ouamour
siham.ouamour@uni.de

Halim Sayoud
halim.sayoud@uni.de