Complex Disfluencies Processing in Spontaneous Arabic Speech

Labiadh Majda[1], Bahou Younès[1,2] and Maaloul Mohamed-Hédi[1,2]
[1]Sfax University, MIRACL Laboratory, Sfax, Tunisia
[2]Hail University, Hail, Saudi Arabia

Abstract. In this paper, we propose a numerical learning-based method for the automatic processing of complex disfluencies in spontaneous oral Arabic utterances. This method allows, from a pretreated and semantically labeled utterance, to delimit and label the conceptual segments of a processed utterance. Then, it allows, from a segmented utterance, to detect and delimit the disfluent segments and then correct them.

Thus, this work is part of the realization of the Arabic vocal server SARF [Bahou, 2014]. We also implemented our MTDC complex disfluencies processing module. For evaluation, we found satisfactory results with an F-measure equal to 91.9%. After integrating the MTDC module into the SARF system, we found an improvement of 11.88% in the acceptable comprehension and 3.77% in the error rate.

Keywords: Spontaneous Arabic speech, Complex disfluencies processing, numerical learning-based method.

1 Introduction

Several systems fall within the scope of automatic natural language processing namely, machine translation, automatic summarization and human-machine oral dialogue systems. We are interested in this article in the automatic understanding of spontaneous oral Arabic utterances. Indeed, the goal of the automatic understanding is to solve several problems. These difficulties are generally due to voice recognition errors and spontaneity errors such as the phenomenon of disfluency (self-correction, repetition, etc.).

Thus, our work consists in proposing a method for the automatic processing of complex disfluencies (complex self-corrections and complex repetitions) to allow the understanding of the spontaneous speech.

Indeed, our research work is part of the realization of the Arabic voice server of information on rail transport, SARF [Bahou, 2014], which is a work on the automatic understanding of spontaneous oral Arabic realized in our laboratory, MIRACL. Thus, we aim to test the performance of SARF by applying a numerical learning-based approach instead of the symbolic approach already in effect.

This article is organized around five sections. The first section will be devoted to the presentation of existing complex disfluency processing approaches. We will present, in the second section, the method for which we opted for the processing of complex disfluencies. In a third section, we will evaluate our module, which has been set up for

the processing of complex disfluencies. Then, in the fourth section, we will compare two versions of SARF system (with or without MTDC module). Finally, we will call to mind our main conclusions while highlighting the contribution of this approach.
We will begin by presenting, in the following section, the processing approaches of complex disfluencies.

2 Processing approaches of complex disfluencies

Several methods were proposed for complex disfluencies processing. In this section, we propose to dwell on these methods. To address them, we chose to classify them into three approaches that we will present respectively as follows: the symbolic approach, the numerical learning-based approach and the hybrid approach.

The symbolic approach is based on a syntactic and/or semantic analysis of the utterance. Several systems use this approach. Indeed, as part of speech processing, Koushik et al. [Kaushik et al., 2010] developed algorithms to effectively suppress the pauses and repetitions of spontaneous speech. Also, as part of the understanding of spontaneous Arabic speech, Bahou et al. [Bahou et al., 2010] proposed a method for the disfluencies processing.

The numerical learning-based approach extracts a set of knowledge from a large amount of data. The numerical approach is generally more robust regarding the difficulties of the spontaneous oral utterances. In addition, these methods are easily transferable from one application or from one language to another.

Among the works carried out on this approach we quote the work of Qader et al. [Qader et al., 2017]. They presented an innovative formalization of revisions, repetitions and pauses in order to allow the automatic addition of disfluencies in the input utterance of a speech synthesis system. In parallel and similarly, we find the work of Christodoulides and Avanzi [Christodoulides and Avanzi, 2015], who presented a detailed annotation scheme and a modular automatic detection system for disfluencies such as pauses, repetitions and false-starts, targeting the semi-automatic annotation of these phenomena in manually-transcribed data of a spoken corpus in French.

We note, however, that the work and method we propose are based on this approach. In fact, we concluded, after some documentation and research, that this approach has not addressed in terms of automatic understanding of spontaneous Arabic speech.

The hybrid approach is a method that combines both symbolic and numerical learning-based approaches to benefit their advantages and avoid their disadvantages.

Among the systems that based on this approach, we quote the work of Proença et al. [Proença et al., 2017]. They developed a method based on two steps namely, a segmentation step and a disfluency-detection step. In addition, as part of the automatic understanding of spontaneous Arabic speech, the conceptual segmentation module of Arabic, AOCSM oral utterances ([Abbassi et al., 2017], [Bahou et al., 2017][a]), based on a symbolic approach (using a set of rules for the conceptual segmentation and the disfluencies processing) and the numerical approach.

3 Proposed method

In this section, we propose a numerical learning-based method for complex disfluencies processing. In fact, we aim to improve the module of the literal comprehension of the SARF system by applying a numerical learning-based method instead of the symbolic method currently adopted in SARF system.
The proposed method allows, from a pretreated and semantically-labeled utterance, to delimit and label the conceptual segments of a processed utterance. Then, it allows, from a segmented utterance, to detect and delimit the disfluent segments and then correct them. Figure 1 describes the steps of the proposed method.

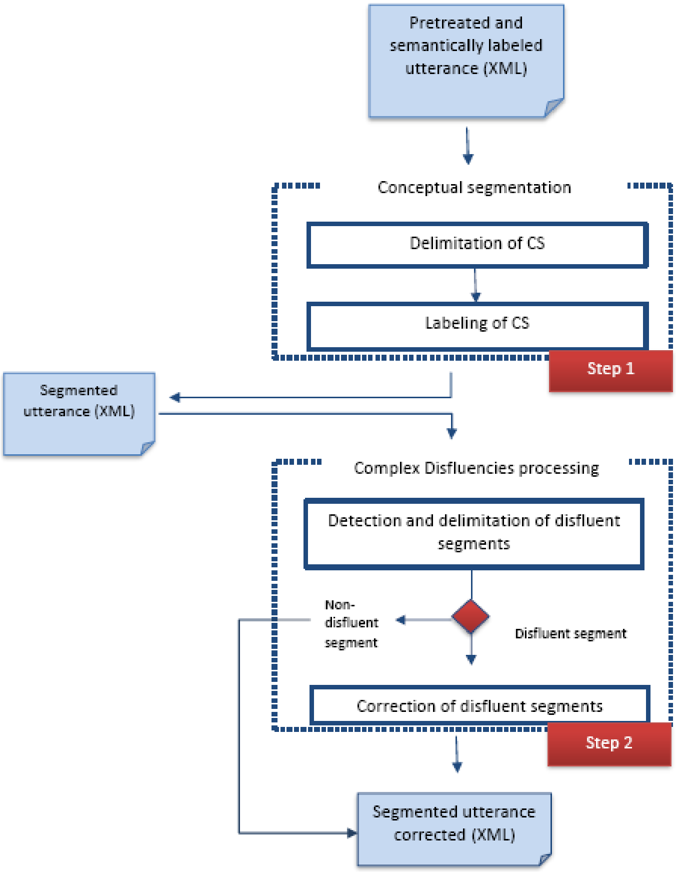

Pretreated and semantically labeled utterance (XML)
Conceptual segmentation
Delimitation of CS
Labeling of CS
Step 1
Segmented utterance (XML)
Complex Disfluencies processing
Detection and delimitation of disfluent segments
Non-disfluent segment
Disfluent segment
Correction of disfluent segments
Step 2
Segmented utterance corrected (XML)

Fig.1. Steps of the proposed method

In what follows, we will focus on these steps (conceptual segmentation and complex disfluencies processing).

3.1 Conceptual segmentation

Conceptual segmentation consists of two sub-steps namely, the conceptual segments delimitation and the conceptual segments labeling. The latter is based on a numerical learning-based approach ([Boughariou et al., 2017], [Bahou et al., 2017][b]). To explain these sub-steps, we propose the utterance (1) as an example.

كم ثمن تذكرة ذهاب الى قابس تذكرة ذهاب اياب الى قابس (1)
[qAbs AlY AyAb *hAb t*krp qAbs AlY *hAb t*krp vmn km][1]
How much is a one-way ticket to Gabes, a return ticket to Gabes

Delimitation of conceptual segments

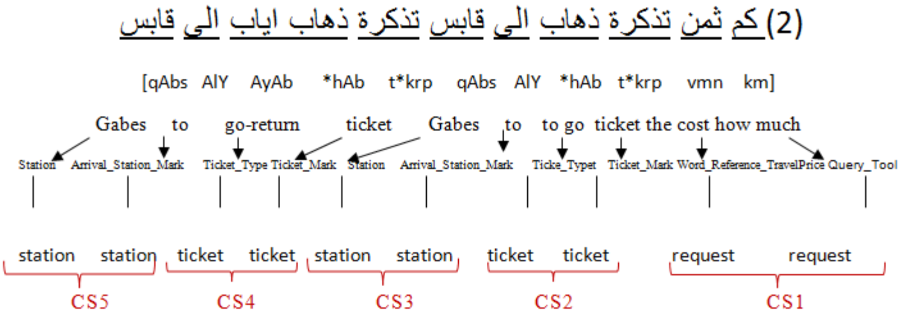
The first sub-step is to extract the conceptual segments of the utterance without resorting to labeling. In fact, any word can have one of the following classifications: "new" (if the word is a triggering index of a new segment), "upstream" (if the word belongs to the previous segment) or "Isolated" (if the word does not belong to any segment). To classify the words of the utterance, we use the following learning criteria:

Table 1. Learning Criteria for Delimitation of Conceptual Segments

Criteria	Value Space	Comments
Tag_Word, Class_Word	Label and semantic class	Semantic tag and class of the candidate word
Tag_PW, Tag_FW	Semantic label	Semantic tags of the predecessor and the successor of the candidate word
Class_PW, Class_FW	Semantic class	Semantic classes of the predecessor and the successor of the candidate word
If_Request	Boolean	Testing if the word is tagged ‘Request’
If_Mark	Boolean	Testing if the word is tagged ‘Mark’
If_PW_Request	Boolean	Testing if the previous word is tagged ‘Request’
Length_Utt	Numerical	Length of the utterance
If _PW_Mark	Boolean	Testing if the previous word is tagged ‘Mark’
Rectif_Mark	Boolean	Testing if the utterance contains a rectification mark

[1] The transliteration of the Arabic examples is based on *Buckwalter*.

To find the vectors of learning, we apply the learning criteria presented above to each word of the utterance (1). However, the set of learning vectors forms an input file for the learning algorithm. In this sub-step, we choose the J48 learning algorithm.
After applying the rules generated by the learning phase, we obtained the conceptual segmentation of the utterance (1), hence the utterance (2) which is only the result of the delimitation of the conceptual segments of the utterance (1).


(2) كم ثمن تذكرة ذهاب الى قابس تذكرة ذهاب اياب الى قابس
[qAbs AlY AyAb *hAb t*krp qAbs AlY *hAb t*krp vmn km]
Gabes to go-return ticket Gabes to to go ticket the cost how much
Station Arrival_Station_Mark Ticket_Type Ticket_Mark Station Arrival_Station_Mark Ticke_Typet Ticket_Mark Word_Reference_TravelPrice Query_Tool
station station ticket ticket station station ticket ticket request request
CS5
CS4
CS3
CS2
CS1

Labeling of conceptual segments

The second sub-step is to detect the type of each conceptual segment extracted in the previous sub-step. Then, to extract the types of each segment, we use the following learning criteria:

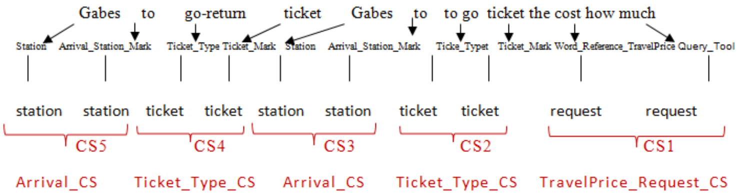
Table 3. Learning Criteria for Conceptual Segment Type Detection

Criteria	Value Space	Comments
Length_CS	Numerical	The conceptual segment length
Pos_CS	Numerical	The conceptual segment position in the utterance
FistW_Pos,LastW_Pos	Numerical	First word and last word positions in the conceptual segment
FistW_Tag,LastW_Tag	Semantic label	First word and last word tags in the conceptual segment
Mark_type	Semantic label	Type of the mark in the conceptual segment
Sem_Class	Semantic class	Semantic class having the best score in the semantic classes
If_DepMark	Boolean	Testing if the conceptual segment contains a departure mark
If_ArrMark	Boolean	Testing if the conceptual segment contains an arrival mark

Thus, each conceptual segment will be notified by its learning vector by applying the preceding criteria. In this sub-step, we chose the J48 learning algorithm.
From the above, we conclude that the J48 algorithm gives the best result. After applying the rules generated in this learning, we came to detect the labels of each conceptual segment, hence the utterance (3), which is the result of the labeling of the conceptual segments of the utterance (2).

(3) كم ثمن تذكرة ذهاب الى قابس تذكرة ذهاب اياب الى قابس

[qAbs AlY AyAb *hAb t*krp qAbs AlY *hAb t*krp vmn km]



3.2 Complex disfluencies processing

The complex disfluencies processing composed of two main sub-steps namely, the sub-step of detecting and delimiting the disfluent segments and the sub-step of their corrections.
To explain these two sub-steps, we will use the labeled and segmented utterance (3).

Detection and delimitation of disfluent segments

This sub-step is essentially based on a numerical learning-based approach to detect complex disfluencies (complex self-corrections and complex repetitions) in spontaneous oral Arabic utterances. To explain this sub-step, we take the utterance (3) as supporting evidence.
This sub-step consists in delimiting and detecting the disfluent segments in the utterance. In fact, each combination must be classified as "None" (if there is no disfluency), "complex self-correction" or "complex repetition".
First, we treat all the possible combinations between the conceptual segments (see table 5).

Table 5. Possible Combinations Between Conceptual Segments

Combinations		Description
Combination 1 :	CS1+CS2	كم ثمن تذكرة ذهاب
Combination 2 :	CS1+CS3	كم ثمن الى قابس
Combination 3 :	CS1+CS4	كم ثمن تذكرة ذهاب اياب
Combination 4 :	CS1+CS5	كم ثمن الى قابس
Combination 5 :	CS2+CS3	تذكرة ذهاب الى قابس
Combination 6 :	CS2+CS4	تذكرة ذهاب تذكرة ذهاب اياب
Combination 7 :	CS2+CS5	تذكرة ذهاب الى قابس
Combination 8 :	CS3+CS4	الى قابس تذكرة ذهاب اياب
Combination 9 :	CS3+CS5	الى قابس الى قابس
Combination10 :	CS4+CS5	تذكرة ذهاب اياب الى قابس

Secondly, to classify these combinations, we use a set of learning criteria (see table 6).

Table 6. Learning Criteria for the Detection and Delimitation of Disfluent Segments

Criteria	Value Space	Comments
Num_Segments	{1,2,3}	Number of Segment Components
Pos_Combination	{1,2,3}	Combination Position
ExistMarkRectif	Boolean	If there is a rectification marker
If_Identique	Boolean	If CS1=CS2
Segment_Label	Belongs to {list of semantic tags}	Segment label
If_identique_label	Boolean	If label of CS1= label of CS2
If_Identique_Part	Boolean	If both segments contain labels of the same type

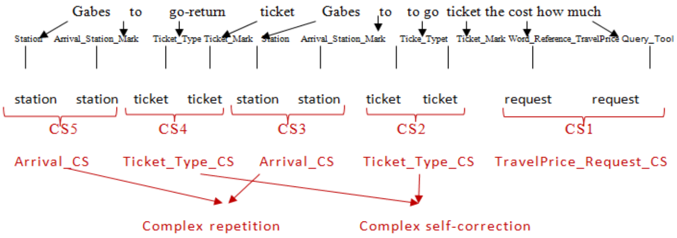
Also, we apply these criteria to all combinations to obtain the learning vectors. The learning vectors obtained form, in fact, an input file for the learning algorithm. For this purpose, we applied the SVM algorithm.

The result of this learning classifies the combination «تذكرة ذهاب اياب تذكرة ذهاب» [AyAb *hAb t*krp *hAb t*krp] (go-return ticket one-way ticket) as a complex self-correction and classifies the combination «الى قابس الى قابس» [AyAb *hAb t*krp *hAb t*krp] (Gabes to Gabes to) as a complex repetition.

The utterance (4) is the result of detecting and delimiting the disfluent segments of the utterance (3).

(4) كم ثمن تذكرة ذهاب الى قابس تذكرة ذهاب اياب الى قابس

[qAbs AlY AyAb *hAb t*krp qAbs AlY *hAb t*krp vmn km]



After this sub-step, we proceed to the second sub-step of the correction of disfluent segments.

Correction of disfluent segments

This step consists in correcting the utterance which contains the disfluencies, or more precisely, disfluent segments. For correction, the disfluent segment is described as follows:

- *Reparandum* (the segment that will be corrected later).
- Optional *Interregnum* (e.g. the rectification marker).

- *Repair* (the part that corrects the *Reparandum*).

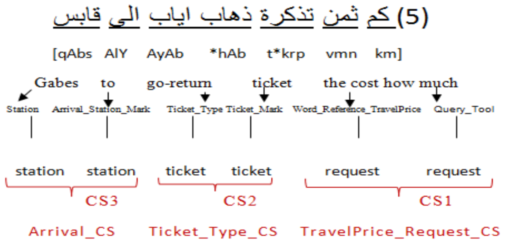
The correction of complex disfluencies of the utterance (4) is as follows:
Complex self-correction:

تذكرة ذهاب اياب   تذكرة ذهاب
[AyAb *hAb t*krp]   [*hAb t*krp]
(go-return ticket)   (to go ticket)
Repair   *Reparandum*

Complex repetition: In complex repetition, the *Repair* and the *Reparadum* are identical.

الى قابس   الى قابس
[qAbs AlY]   [qAbs AlY]
(Gabes to)   (Gabes to)

Subsequently, we replaced the *Reparandum* with the *Repair*. The utterance (5) is the utterance (4) after correction.



4 Evaluation of MTDC module

To properly evaluate the MTDC module, we used some evaluation measures namely, Precision, Recall and F-measure. To this end, we compared the results of MTDC module with the results found and concluded by the linguistic experts.
In our evaluation, the Recall measure represents the number of correctly-corrected disfluencies compared to the number of disfluencies found by the system. On the other hand, the Precision measure represents the number of correctly-corrected disfluencies in relation to the number of disfluences to be found.
We found satisfactory results with an F-measure equal to 91.9%. The table 8 illustrates the results obtained.

Table 8. Evaluation of MTDC Module

	Recall	Precision	F-measure
MTDC module	89.14%	95%	91.9%

5 Comparison with the SARF system

We have integrated our MTDC module into the literal comprehension module of the SARF system. In fact, we generated the semantic frames. First, the generation of semantic frames consists in identifying the semantic frame that corresponds to the reference words (ثمن، وقت، مدة). Secondly, it allows filling this frame based on the information extracted from the conceptual segments.

After integrating the MTDC module into the SARF comprehension module, we compared the initial version with the enhanced version of the SARF comprehension module. We further specify that the SARF evaluation was calculated considering the three scores proposed by [Bahou, 2014] namely, Complete Comprehension CC, Incomplete Comprehension CI and Erroneous Comprehension EC.
Thus, (CC + CI) indicates the number of utterances with an acceptable literal comprehension. (CI + EC) indicates the error rate of literal comprehension.

Table 8. Comparison Between the Two Versions

	CC	CI	EC	CC+CI	CI+EC
Initial version	75.53 %	6.91%	17.55%	82.44%	24.46%
Improved version	79.29%	15.03%	5.66%	94.32%	20.69%

After this evaluation, we found an improvement of 11.88% in the acceptable comprehension and 3.77% in the error rate. Indeed, this improvement is mainly due to the application of a numerical learning-based approach unlike that used by SARF and which based on a literal comprehension module.

6 Conclusion

In this article, we have proposed a method for the automatic processing of complex disfluencies in spontaneous oral Arabic utterances. This numerical method adopts the learning technique for conceptual segmentation and detection and delimitation of disfluent segments. The proposed method is part of the realization of the Arabic SARF vocal server.
The evaluation of our realized module is satisfactory with an F-measure equal to 91.9%. Thus, after integration with the SARF, we noticed an improvement of 11.88% in the acceptable comprehension and 3.77% in the error rate.

References

1. Abbassi H., Bahou Y., Maaloul M-H. : L’apport d’une approche hybride dans la compréhension de l’oral arabe spontané. 29th of Proceedings of International Business Information Management Association (IBIMA'17), Vienna, Austria, (2017).

2. Bahou Y. : Compréhension Automatique de la Parole Arabe Spontanée : Intégration dans un Serveur Vocal Interactif. PhD Thesis, Faculté des Sciences économiques et de Gestion de Sfax, (2014).
3. *(a)* Bahou Y., Maaloul M-H., Abbassi H. : Hybrid approach for conceptual segmentation of spontaneous Arabic oral utterances, 3rd International Conference on Arabic Computational Linguistics (ACLing'17), Dubai, UAE, (2017).
4. *(b)* Bahou Y., Maaloul M-H., Boughariou E. : Towards the supervised machine learning and the conceptual segmentation technique in the spontaneous Arabic speech understanding. 3rd International Conference on Arabic Computational Linguistics (ACLing'17), Dubai, UAE, (2017).
5. Bahou Y., Masmoudi A., Hadrich-Belguith L. : Traitement des disfluences dans le cadre de la compréhension automatique de l'oral arabe spontané. 28èmes Journées d'Études sur la Parole (JEP'10), Mons, Belgique, (2010).
6. Boughariou E., Bahou Y., Maâloul M-H, : Application d'une méthode numérique à base d'apprentissage pour la segmentation conceptuelle de l'oral arabe spontané", of Proceedings of International Business Information Management Association (IBIMA'17), Vienna, Austria, (2017).
7. Christodoulides G., Avanzi M. : Automatic Detection and Annotation of Disfluencies in Spoken French Corpora. Institute for Language & Communication, University of Louvain, Belgium DTAL, Faculty of Modern &Medieval Languages, University of Cambridge, UK, (2015).
8. Kaushik M., Trinkle M., Hashemi-Sakhtsari A. : Automatic detection and removal of disfluencies from spontaneous speech. In School of Electrical and Electronic Engineering, The University of Adelaide, Adelaide, South Australia. C3I Division, Defense Science and Technology Organization, Edinburgh, South Australia, (2010).
9. Proença J., Lopes C., Tjalve M., Stolcke A., Candeias S., Perdigão F. : Automatic evaluation of reading aloud performance in children. Department of Electrical and Computer Engineering, University of Coimbra, Portugal, (2017).
10. Qader R., Lecorvé G., Lolive D., Sébillot P. : Ajout automatique de disfluences pour la synthèse de la parole spontanée : formalisation et preuve de concept. Traitement Automatique du Langage Naturel (TALN'17), Orléans, France, (2017).