Summarizing videos into a target language: Methodology, architectures and evaluation

Kamel Smaïli, Dominique Fohr, Carlos-Emiliano González-Gallardo, Michal L Grega, Lucjan Janowski, Denis Jouvet, Arian Koźbial, David Langlois, Mikolaj Leszczuk, Odile Mella, et al.

Summarizing videos into a target language: methodology, architectures and evaluation

K. Smaïli[1], D. Fohr[1], C.E. González-Gallardo[2], M. Grega[3], L. Janowski[3], D. Jouvet[1], A. Kozbiał[3], D. Langlois[1], M.Leszczuk[3], O. Mella[1], M.A Menacer[1], A. Mendez[4], E.L. Pontes[2], E. SanJuan[2], J.M. Torres-Moreno[2], and B. Garcia-Zapirain[4]

[1] Loria, University of Lorraine, France
{smaili, fohr, jouvet, langlois, mella, mohamed-amine.menacer}@loria.fr
[2] LIA, Avignon Université, France {carlos-emiliano.gonzalez-gallardo, Elvys-linhares.pontes, eric.sanjuan, juan-manuel.torres}@univ-avignon.fr
[3] AGH University of Science and Technology Krakow – Poland
{grega, janowski,kozbial,Leszczuk }@kt.agh.edu.pl
[4] University of DEUSTO Bilbao – Spain
{amaia.mendez,mbgarciazapi}@deusto.es

Abstract. The aim of the work is to report the results of the Chist-Era project AMIS (Access Multilingual Information opinionS). The purpose of AMIS is to answer the following question: How to make the information in a foreign language accessible for everyone? This issue is not limited to translate a source video into a target language video since the objective is to provide only the main idea of an Arabic video in English. This objective necessitates developing research in several areas that are not, all arrived at a maturity state: Video summarization, Speech recognition, Machine translation, Audio summarization and Speech segmentation. In this article we present several possible architectures to achieve our objective, yet we focus on only one of them. The scientific locks are be presented, and we explain how to deal with them. One of the big challenges of this work is to conceive a way to evaluate objectively a system composed of several components knowing that each of them has its limits and can propagate errors through the first component. Also, a subjective evaluation procedure is proposed in which several annotators have been mobilized to test the quality of the achieved summaries.

Keywords: Automatic Speech Recognition Statistical Machine Translation · Video Summarization · Audio summarization · Objective and Subjective evaluation

1 Introduction

When we want to access information, the first reflex is to turn to the Internet to access useful knowledge. Information on the Internet is diverse and presented in different formats and various languages. The language barrier prevents people from accessing the huge amount of knowledge and especially their diversity. Multimedia platforms like YouTube offer an automatic translation service for some videos, however two problems must be mentioned about this service. The first problem concerns the quality of the translation while the second, the fact that a foreign user sometimes wishes to have just the main idea of a video in his own language presented in only few seconds.

One of the objective of this project is to make accessible some information presented in a foreign language. The consequence of getting this information is to be aware about the existence of another version of a topic or to get another sound of a story. For instance, the AIDS topic is not performed in the same way by the journalists in the West or by those of an Arabic country for different socio-cultural reasons. To do so, AMIS (Access Multilingual Information OpinionS), a Chist-Era project[5] helps people to retrieve from a video the main topic by summarizing the original video in a target language and in the desired duration. To realize such a system, several knowledge are needed: video, audio and text summarization, automatic speech recognition and automatic translation. Moreover, it is not enough to nest these components in each others, but a global framework allowing an efficient communication between the different stones of AMIS is needed to propose.

In what follows, we will present the different version of AMIS architectures and evaluate some of them in an objective and subjective way. Each component will be described and the models used will be discussed. The challenges and the scientific locks will be mentioned. To our knowledge, there is no equivalent work, so it will be difficult to compare our results to others.

2 Different components of AMIS

2.1 Video Summarization

We designed and developed an operational framework for summarizing newscasts and reports [16]. The structure is composed in such a form, that it enables for pure experimentation with various methods to summarize video. The framework hosts several high- and low-level meta-data extraction algorithms (referring to our previous research conducted within the scope of, e.g. IMCOP project [1]) that include detection of the anchor-person, identification of daytime and nighttime shootings and extraction of low-level video quality indicators.

The main summarization processes start with Shot Boundary Detection (SBD). This algorithm helps in prediction whether the video is static or dynamic. Also, through SBD we can calculate and compare data per shots instead of frames which is a way more efficient while analyzing video clips over a longer duration. We used the video quality indicators mentioned above for calculating the coefficient of activity which is a product of two indicators – Spatial Activity and Temporal Activity. These indicators show that the number of details appears on the frame and how dynamic the frame is in comparison to the previous frame, respectively. The coefficient of activity is calculated in two steps, first per frame and then as an average per shot. We build the final summary from the shots with a higher or equal coefficient of activity value compared to the average value of the entire video. We illustrate the process of summarizing video sequences in [16].

[5] `http://deustotechlife.deusto.es/amis/`

2.2 Automatic Speech recognition system

Arabic, English and French videos are handled in the AMIS project. A focus is set on the Arabic videos, as they must be summarized in English. For that, the first processing step implies the automatic transcription of Arabic videos. The development of the Arabic speech recognition system relied on state-of-the-art approaches, and large audio and text corpora for taining the models. The system that has been developed was named *ALASR*, for *Arabic Loria Automatic Speech Recognition system* [23]. The acoustic and language models which are the main components are described in the following.

Acoustic model After the parameterisation of the audio signal resulting in a 39-component vector per each frame of 10ms (13 Mel-Frequency Cepstral Coefficients plus their differential and acceleration components), 35 Arabic acoustic models (28 consonants, six vowels and silence) are trained using Hidden Markov Models (HMM) and deep learning. More precisely each non-silence CD-DNN-HMM model consists of a context-dependent 3-state triphone whose the tied states (senones) are the outputs of a deep neural network (DNN). As shown in Figure 1, the neural network is composed of a 440-neuron input layer corresponding of 11 frames of 40 parameters, followed by six hidden layers of 2048 neurons and ended by a softmax layer with 4264 outputs (senones). This topology totals 30.6 million weights to estimate on the Arabic train corpus.

The training of the acoustic models uses the Kaldi toolkit [29]. It is broken down into several training stages: monophonic, triphones using fMLLR for speaker adaptation [6] and LDA to transform $39\times$ 9 features into a vector of 40 components per frame, triphones using adaptative speaker training (SAT), and finally, the DNN part is trained according to the sMBR criterion [36].

Language model To reduce the sparsity of data, the corpus has been preprocessed before estimating the language model [23]. For that, in this work, numbers and dates were converted to words, prefixes were systematically processed (depending on the prefix: concatenation or separation from the following word), duplicated letters expressing strong emotion (for example in English: 'yyyyeeessss') were suppressed and last, shortened forms were replaced with the corresponding sequence of words [23]. Then, the language model has been performed on a corpus composed of two parts: Giga-Word, a huge generic training corpus, and our small corpus made up of speech transcript. Because both corpora are unbalanced, a 4-gram language model was estimated for each part; these both models were linearly combined. Finally, the final language model was pruned by minimising the corresponding entropy among the whole and the pruned model [33].

2.3 Machine Translation

In the scope of the AMIS project, the foreign language is Arabic, and the user speaks English. Therefore, we developed a machine translation system for side Arabic-English. Because the corpus of United Nations organisation is sufficiently broad and diverse, containing parallel sentences for Arabic and English, we decided to train our model

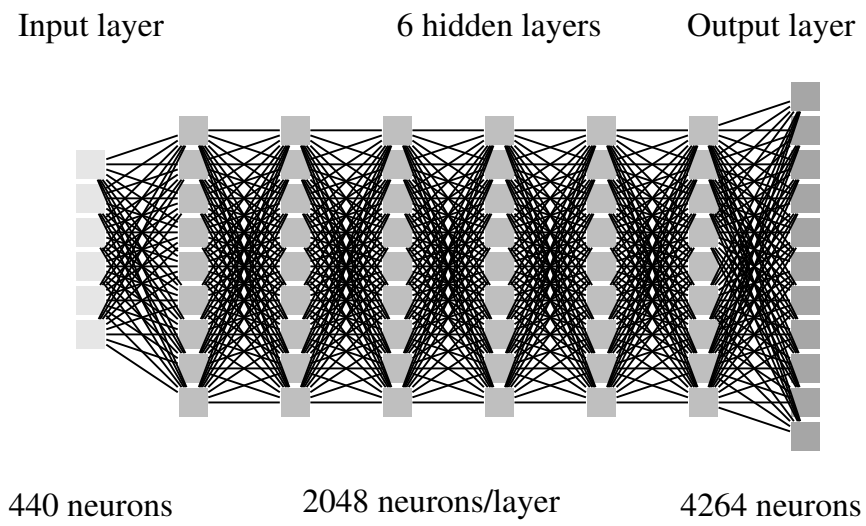

Fig. 1. Acoustic model flow diagram.

on this corpus. We used the data of the period from January 2000 to September 2009 [5]. This resulted in 9.7 million parallel sentences. Some statistics about this corpus are given in Table 1.

Language	# sentences	# words	# unique words
Arabic	9.7M	232.8M	690k
English	9.7M	275.4M	388k

Table 1. Statistics about the parallel corpus for machine translation

Concerning the language model of our MT system, we train a 4-gram language model. Words occurring at least twice were retained and a Kneser-Ney technique [12] has been used for smoothing. Statistics on those data are presented in Table 2.

From the corpus, we also extracted a development and a test corpus; both made up of 3,000 parallel sentences.

To obtain the translation table, we used the classical GIZA++ toolkit [26]. The framework used for translation is MOSES [13]. This system stems on the phrase-based statistical approach [3, 14]. MOSES uses a beam search algorithm and evaluates the translation hypotheses by using a log-linear function. The weights of this function are tuned with the MERT algorithm [27] on a development corpus.

unigrams	225k
bigrams	9.1M
trigrams	13.8M
4-grams	22.5M

Table 2. Number of n-grams in the English language model used for machine translation

2.4 Text Summarization

Automatic Text Summarization (ATS) is an important Natural Language Processing (NLP) task [34]. It aims to find the most relevant information from a document source in order to generate a short informative version. In this project, the text summarizing module aims to produce an abridged version of a newscast or report video based only on the textual information provided by two other modules from the project: the ASR and MT systems. Given the multi-language perspective of the project, the ATS module can produce summaries in French, English and Arabic languages.

ATS can be achieved with three different approaches: extractive, abstractive and sentence compression summarization [34]. Some exploratory experiments with sentence compression summarization have been performed during the project; however, we have mainly developed the extractive summarization approach because of its robustness to external noise like speech disfluencies and ASR mistakes [2, **?**].

ATS depends on the existence of sentences either to select, reformulate or compress the original document. In the AMIS project, the source text to ATS can be an ASR transcript or its translation; neither case contains punctuation marks. Hence sentences are inexistent. To overcome this significant issue, we developed a specialised sentence boundary detection submodule.

Sentence Boundary Detection Different from written text, in spoken language, sentences are not very well defined and have a wider delimitation. In this context, the term sentence-like unit (SU) is used to define each one of the segments within the transcript. Well-formed sentences, phrases and words unigrams can be interpreted as SUs [17]. The sentence boundary detection (SeBD) submodule developed for segmenting ASR transcripts and translations comprise the three languages of the project: English, French and Arabic. The developed SeBD system uses mainly textual features and convolutional neural networks (CNN) to segment the transcripts and generate SUs [7].

2.5 Audio summarization

We addressed audio summarization as a Machine Learning task. Similar to text summarization, audio summarization aims to select those segments of the original video that are more relevant and produce an abridged and informative version of it. During the training phase, a linear regression model is trained to map the given audio features of each segment with an informativity value obtained from the Jensen–Shannon divergence between the segment transcript and the complete audio transcript. In this sum-

marization method, only audio features are used to represent each segment. All audio processing and feature extraction is performed with the Librosa library [6] [22].

Pre-procesing During the pre-processing step, the video's audio signal is split into background and foreground. This process is used on music records for separating vocals and other sporadic signals from accompanying instrumentation. Rafii et al. [31] achieve this separation identifying repeating elements by looking for similarities instead of looking for periodicities by using a similarity matrix. This approach is useful for those song records where repetitions happen intermittently or without a fixed period. However, we found that applying the same method to newscasts and reports audio files made much more comfortable with segmenting them using only the background signal. This is because newscasts and reports are heavily edited with usually a low volume background music playing while the journalist speaks (background) and with louder music/noises for transitions (foreground).

Following [31], to suppress non-repetitive deviations from the average spectrum and discard vocal elements, audio frames are compared using cosine similarity. Similar frames separated by at least two seconds are aggregated by taking their per-frequency median value to avoid being biased by historical continuity. Next, assuming that both signals are additive, a pointwise minimum between the obtained frames and the original signal is applied to obtain a raw background filter. Then, a foreground and background time-frequency masks are derived from the raw background filter and the input signal with a soft mask operation. Finally, foreground and background components are obtained multiplying the time-frequency masks with the input signal. The background audio component is represented with 25 Mel-frequency Cepstral Coefficients (MFCC) features [24] and is segmented into 20 groups per minute with a clustering mechanism.

Audio Summary Creation For each segment, Q within the background audio component P, a S_Q value is computed to rank its pertinence in summary. Audio summarization is performed choosing those segments which contain higher S_Q in order of appearance until a length percentage is reached. S_Q is defined as:

$$S_Q = \frac{1}{1+e^{-(\Delta_t-5)}} \times lr_Q \times e^{-\frac{t_Q}{\Delta_t}} \times e^{1-D_{JS}(P||Q)} \tag{1}$$

Here $\Delta_t = t_{Q+1} - t_Q$, being t_Q the starting time of the segment Q and t_{Q+1} the starting time of the segment $Q+1$. lr_Q is the ratio between the length of the segment Q and the complete audio. $D_{JS}(P||Q)$ corresponds to the Jensen-Shannon divergence between the corresponding segment transcript and the complete audio transcript P as defined in [18, 35]:

$$D_{JS}(P||Q) = \frac{1}{2}\sum_w \left(P_w \log_2 \frac{2P_w}{P_w+Q_w} + Q_w \log_2 \frac{2Q_w}{P_w+Q_w}\right) \tag{2}$$

[6] https://librosa.github.io/librosa/index.html

$$P_w = \frac{C_w^P + \delta}{|P| + \delta \times \beta} \tag{3}$$

$$Q_w = \frac{C_w^Q + \delta}{|Q| + \delta \times \beta} \tag{4}$$

where $C_w^{(P|Q)}$ is the frequency of word w over Q or P. To avoid shifting too much probability mass to unseen events, the scaling parameter δ is set to 0.0005. $|P|$ and $|Q|$ correspond to the number of tokens on P and Q. Finally $\beta = 1.5 \times |V|$, where $|V|$ is the vocabulary size.

3 Global architecture

In order to summarize a video in a target language, four different architectures are proposed. These architectures are presented in Figure 3. In *SC*1, a summary video is created directly without using the audio content of the video. The content of the result of this summary is then transcribed using our speech recognition system (ALASR) and then translated into English. The result is integrated as subtitles to the summarized video. *SC*2 is an original architecture in which a summary is proposed based on the audio part of the original video. The result is then converted into subtitles following the same principles as in *SC*1. *SC*3 and *SC*4 are similar to each other. They take benefit from the result of ALASR system. The result of this step is then a text in Arabic, then the blocks (Machine Translation + Text Summarization) and (Text Summarization + Machine Translation) are respectively performed on *SC*3 and *SC*4.

4 Objective Evaluation

The final system will be evaluated globally. However, during the development phase, an evaluation of each component is mandatory to analyse the strengths and weaknesses of each of them. In the next sections, we will evaluate each component individually.

4.1 Evaluation of Video Summarization

In this subsection, we show that it is possible to find methodologies that allow us to avoid personal testing of various approaches for generating summaries. We present two algorithms that can help in evaluating multilingual multimedia summaries.

The first method uses annotation technique based on expert's selection to create a reference summary. Afterwards, all summaries created by summarization scripts or tools are referred to this reference. This algorithm lets checking if a specific summarization approach selected stunning shots from the original video sequence in summary.

The second method is related to YouTube tags originally added to video sequence data. It allows us checking another aspect – the video content. Evaluating video sequences cannot be reduced only to the visual layer. Checking the body (saved as text)

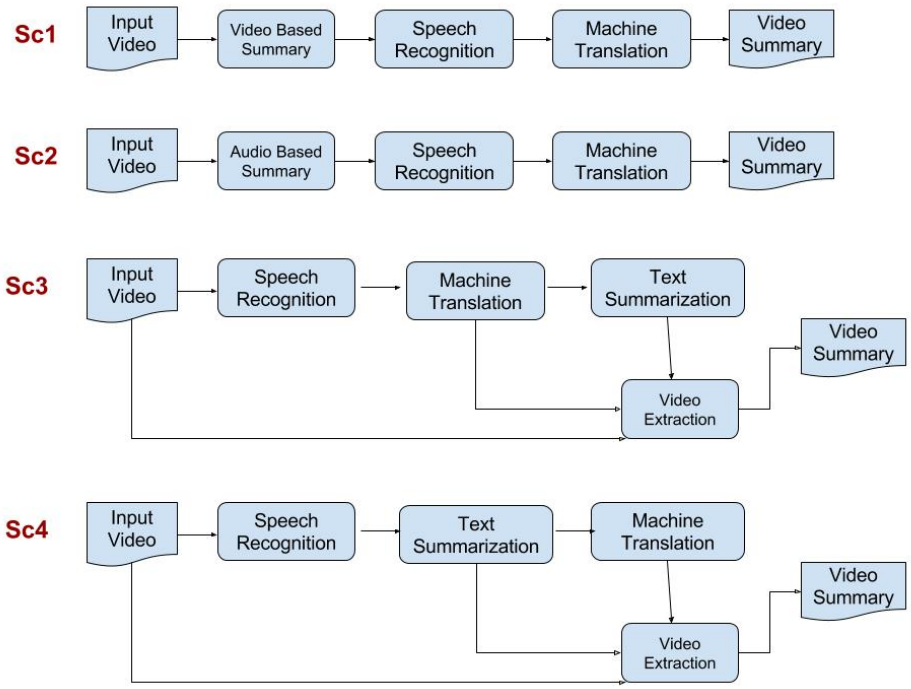

Fig. 2. Different architectures for summarizing a video to a target language

of a video sequence with the use of tags let us check if a summary contains all critical data. Combination of these methods reduces the necessity of subjective tests and lets us evaluate summaries in a sophisticated way.

There are some solutions for a summary evaluation that we found in related research works. Natural Language Descriptions [11] also used this solution. However, in our case, annotating was not as detailed. Instead, we complemented our method with the tag-based method. It is also necessary to mention about Recall-Oriented Understudy for Gisting Evaluation (ROUGE) [28]. We used a few of its concepts including Precision, Recall and F1 score. We also refer to text summary evaluation, which was used at Document Understanding Conference (DUC) [25], and defines that the primary metric is coverage. We partially use this concept while checking if all tags used in the original video sequence are "covered" in a video summary. In [20] we can also find other methods for summary evaluation that inspired us. These are intrinsic and extrinsic approaches, for which, we describe and rate methods for assessing informativeness. This paper is also partially based on concepts given in [21].

Experiments on both evaluation methods used the same set of a test of 41 video sequences. We selected them from YouTube newscast programs and reports presented in three languages: English, French and Arabic. Their duration was from 3 to 13 minutes.

The idea behind selecting these video sequences was to have a similar number of video sequences in three duration intervals (values in minutes): $[3-6]$, $(6-10]$ and $(10-13]$. Finally, we decided to select a little bit more long video sequences (as for them, summarization makes more sense). For the full duration distribution, please check Figure 3.

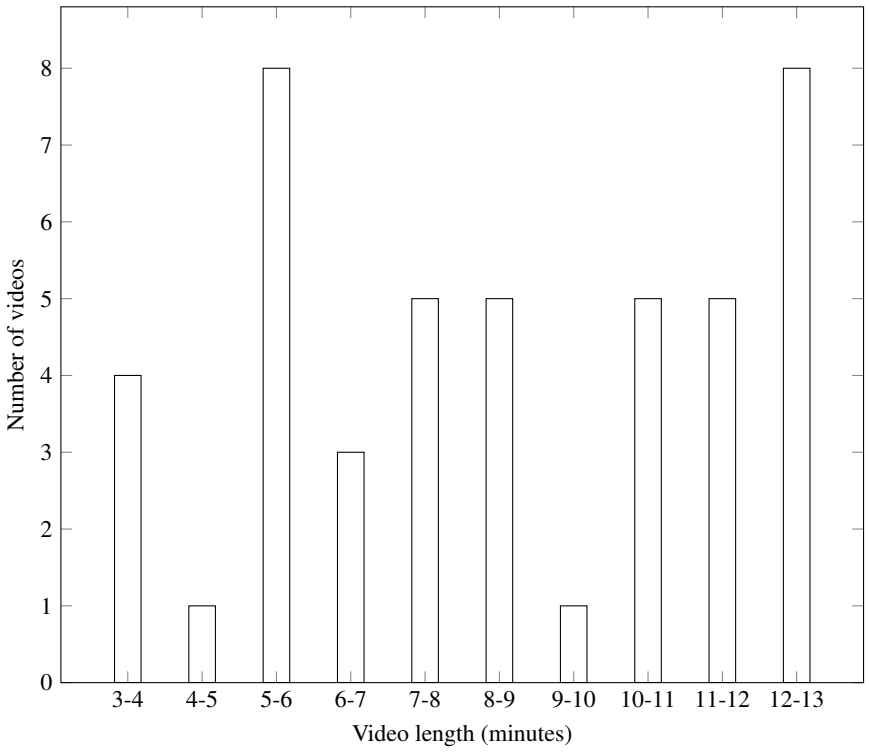

Fig. 3. Video sequence duration distribution.

We processed all data in both methods and stored in a database. It made it easier to retrieve it for further processing.

Annotation Method The framework is evaluated using annotated video sequences. A pool of experts decides which frames are keyframes (meaning: critical frames, the core of the video) and which have to be in summary. We use VLC media player[7] to extract frames from single shots. This evaluation process of choosing keyframes is time-consuming and subjective. In order to describe the obtained results, we are calculating Precision, Recall and F1 score for each sequence and algorithm.

We considered a process of summarizing video based only on visual evaluation. Video sequences were provided both to human and algorithm without any additional

[7] https://www.videolan.org/vlc/index.html

audio description. Of course, a person creating the evaluation could understand some written text appearing on the screen. Our goal was to validate if the summary created by a human is similar to the summary created by an algorithm, focusing on the visual part only. Such evaluation can be found in literature [8, 9, 30, 32] just to name a few. Nevertheless, the literature did not consider the summary of the news. In our research, we noted that making a reasonable summary for the human observer is very difficult. As a consequence comparing to an algorithm is not as precise as for other cases considered in the literature.

The first problem we found is the length of the summary. Table 3 presents the length of summaries provided by a human observer. We can see that the shortest is just 15% of the original video while the longest is 61%. Comparing such different solutions is difficult. The difference comes from the very different nature of the video news, which can span from a talking head to a report from a field where there is an action. In order to help with comparing human-made and automatic video summaries, the automatic algorithm has the information about the length of the summary provided by humans.

Video ID	Summary length	Source length	Percentage
1	205	473	43%
2	86	187	46%
3	76	277	27%
4	69	200	35%
5	85	186	36%
6	119	194	**61%**
7	41	281	**15%**
8	41	233	18%

Table 3. Summary length comparison. A human creates the summaries in this table.

Precision and Recall metrics compared the automatic and human summaries. We calculated how an algorithm also marked many frames marked by a human (we give details of the procedure in [15]). So "true positive" means that both human and algorithm marked the same frame. "True negative" means that both human and algorithm did not mark a specific frame. Table 4 presents the results obtained for all evaluated sequences. The obtained results are not very good but even comparing summaries provided by two humans are not much better. The problem is the content, already included in a summary.

Precision	Recall	F1	Accuracy
0.36	0.13	0.19	0.36

Table 4. Performance of video summarization

Tag-Based Method This paragraph presents the evaluation method based on tags. It contains a data format, the scheme of the used methodology and the results of our experiments.

For the tag-based evaluation method, the most important data used were YouTube tags. We used YouTube tags for every selected video sequence. Tags used there to summarize the given newscast/report video sequence. Tags are, depending on the particular video sequences, written in different languages. Here are some examples:

RT, Russia Today, FSA, war, troops, us-backed
FSA kicks out US special forces troops, Syria
Free Syrian Army, US special forces troops, kicked out

Every video sequence used has its own set of tags. They are in different languages. The algorithm for each video sequence is:

1. Retrieve the audio track from the original video sequence.
2. Use the developed automatic speech recognition system engine to obtain a textual transcription of the original video sequence.
3. Retrieve tags. If a tag includes more than one word, split it. Create a set without duplicates.
4. Check which tags appear in the textual transcription of the original video sequence. Limit the set of tags to these tags that occur in the textual transcription of the original video sequence.
5. Create a summary of the original video sequence.
6. Retrieve an audio track from the recently created summary.
7. Use the ASR engine to obtain a textual transcription of the summarized video sequence.
8. Check which tags appear in the textual transcription of the summarized video sequence. Create a set of tags that occur in the textual transcription of the summarized video sequence.
9. Check the tags that occur in the summarized and the original video sequence. Calculate statistics.

In Figure 4 one can observe the number of tags in the original video sequences and the number of tags in the summarized video sequences – for all tested video sequences.

As we can see, for most of the selected video sequences, the percentage of occurring tags in summaries is above 50. It means that, despite different relative lengths of the summarized video sequences for various original video sequences, they contain content described in tags. It is a valuable check; however, to get more reliable information, a test should be done for a broader set of video sequences.

4.2 Evaluation of Arabic Automatic Speech Recognition

The ASR system for Arabic has been trained on an acoustic corpus of 63 hours (Nemlar [19] and NetDC [4]). The Language Model has been trained on the GigaWord. The vocabulary is composed of 95k words with an average of 5.07 pronunciations for each entry. After several tests, tuning and improvements ALASR achieves the performance

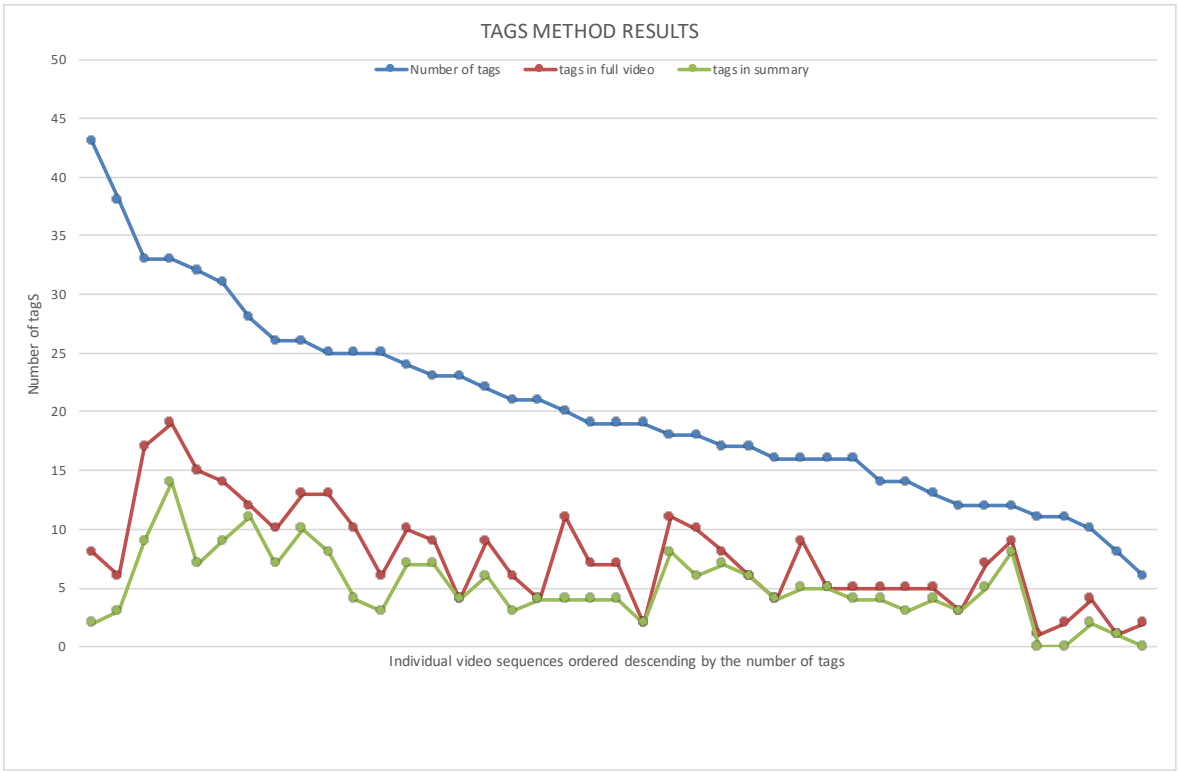

Fig. 4. Tag-based evaluation method results

presented in Table 5. This performance is achieved on a tuning and test corpora of 31,000 sentences for each of them. This test is done on data not extracted from our video database. The issue is that we do not have any reference transcription corpus for these videos; hence, evaluation is impossible.

To overcome this problem, we decided to build a pseudo-reference by aligning, for each YouTube video from Euronews channel, the automatic transcript and textual data from the corresponding Youtube and Euronews webpages. The transcript is considered as a reference if the WER is under a chosen threshold [10]. Experiments have been done on a corpus of 1300 sentences (a mixture of transcripts from YouTube and Euronews). We have to notice that this transcript does not correspond exactly to what has been pronounced. Consequently, the performance we provide below is under-estimated. Under these conditions, ALASR achieves a WER of 36.5.

	Dev WER	Test WER
ALASR	13.07	14.02

Table 5. Performance of ALASR in terms of WER

4.3 Evaluation of machine translation

We first evaluate the machine translation system on a corpus of 3000 sentences extracted from the United Nations (UN) (Table 6) by using several standard evaluation measures: BLEU, METEOR, TER, and WER, all ranging between 0 and 1.

	Test (3k sentences)
BLEU	0.39
METEOR	0.29
TER	0.56
WER	0.62

Table 6. The evaluation of the Arabic–English MT system on the UN test set

As for the ASR system, the evaluation on the database we collected is not easy since we need a reference corpus. Unfortunately, this reference corpus is not available. In order to have an idea about the relevance of the machine translation system on our database, we create artificially a pseudo-reference corpus by translating with Google-Translate[8] and Systran[9], 197 videos of Euronews that correspond to 1253 sentences. The results are given in Table 7, in terms of BLEU (high values are a better performance).

System	**AMIS**	**Google**	**Systran**
AMIS	–	26.7	9.9
Google	26.7	–	12.8
Systran	10	12.9	–

Table 7. The evaluation of the Arabic–English MT system on AMIS data.

The BLEU for our system on the pseudo-reference corpus achieved by Systran (PRS) is weak, only 9.9, while the performance on the pseudo-reference corpus obtained by Google-Translate (PSG) is equal to 26.7. To understand the weak performance on PRS, we launched Google-Translate on PRS. The achieved BLEU for this experience is 12.8. This illustrates that both Google and our system fail to get good

[8] https://cloud.google.com/translate/
[9] http://www.systransoft.com/

performance on PRS. This is probably due to the fact that the translation with Systran is not good on the transcription of the videos.

4.4 Evaluation of Sentence Boundary Detection (SeBD)

This submodule performs a binary classification that decides whether a target word corresponds to a boundary between two SUs. Table 8 presents our results of a strict evaluation for the Arabic, French and English SeBD submodule in terms of Precision (P), Recall (R) and their harmonic mean ($F1$).

Evaluation was performed over 12 million samples from the *Arabic Gigaword Fourth Edition*, 106 million samples from the *English Gigaword Fifth Edition* and 117 million samples from the *French Gigaword First Edition*. In general, the SeBD submodule achieved good results concerning the "no boundary" class; both P and R are over 92%. However, the performance related to the "boundary" class dropped almost 15% for P and 32.5% for R. The unbalanced nature of the data influences this decline in performance. The "no boundary" class represents the majority of samples (84%) while the "boundary" class represents only 16% of samples. Further work is under progress to improve the performance on the "boundary" class.

Language	Class	P	R	$F1$
Arabic	no boundary	0.928	0.963	0.945
	boundary	0.782	0.638	0.700
English	no boundary	0.977	0.980	0.976
	boundary	0.838	0.796	0.816
French	no boundary	0.975	0.986	0.981
	boundary	0.845	0.754	0.795

Table 8. Performance of the AMIS SeBD submodule

4.5 Evaluation of audio summarization

Audio summarization preliminary evaluation was performed over a small set of 10 English videos. Selected videos length varies between 102 seconds (1m42s) and 584 seconds (9m44s) with an average length of 318 seconds (5m18s). Summaries length was set to be the 35% of the original audio length.

Evaluation was performed over the complete audio summaries as well as over each summary segment. For each case informativity was measured with a discrete scale from 1 to 5, going from non-informative to total informative. Table 9 shows the length of each video and the number of selected segments during the summarization process. "Full Score" corresponds to the complete audio summaries score while "Average Score" to the average score of their corresponding summary segments. Both metrics represent different things and seem to be not correlated. Whereas "Full Score" is used to evaluate the informativity of all the summary as a whole, "Average Score" represents the

informativity quality of the summary segments. To validate this observation, we computed the linear correlation between these two metrics obtaining a *PCC* value equal to -0.0267.

The lowest "Full Score" value obtained during the evaluation was 3 and the higher 5, meaning that the summarization algorithm generates at least half informative summaries. "Average Score" values oscillate between 2.90 and 4.14. A compelling case is video #6, which according to its "Full Score" is informative but has the lowest "Average Score" of all samples. This difference is given because 67% of its summary segments has an informativity score of ≤ 3, but in general, it achieves to communicate all the relevant information.

Video	Length	# Segments	Full Score	Average Score
1	3m19s	8	5	3.38
2	5m21s	13	3	3.31
3	2m47s	5	3	4.00
4	1m42s	5	5	3.00
5	8m47s	22	5	4.14
6	9m45s	30	5	2.90
7	5m23s	8	4	3.50
8	6m24s	20	3	2.95
9	7m35s	18	4	3.67
10	2m01s	4	3	3.25

Table 9. Audio summarization performance over complete summaries and summary segments

A graphical representation of the audio summaries and their performance can be seen in Figure 5. Full audio streams are represented as blue bars while summary segments are represented by the grey zones, which height has corresponded to their informativity score.

From Table 9, it can be seen that videos #2, #3, #8 and #10 have full scores of 3, which corresponds to half educational value. As seen in Figure 5, these videos have all their summary segments clustered to the left. This is due to the preference that the summarization technique gives to the first part of the audio stream, a region wherein a standard newscast is gathered the major part of the information. The problem is that in some cases, where different topics are covered over the newscast (multi-topic newscast, interviews, round tables, reports, etc.), the relevant information is distributed all over the video. So, if a significant amount of relevant segments are grouped in this region, the summarization algorithm uses all the space available for the summary very fast, leaving away a vast region of the audio stream. By contrast, video #7, which also has all its summary segments clustered to the left has a full score equal to 4. In this case, the second half of the video does not contain much relevant information; thus, focusing on its first part does not have a significant repercussion on the audio summarization performance.

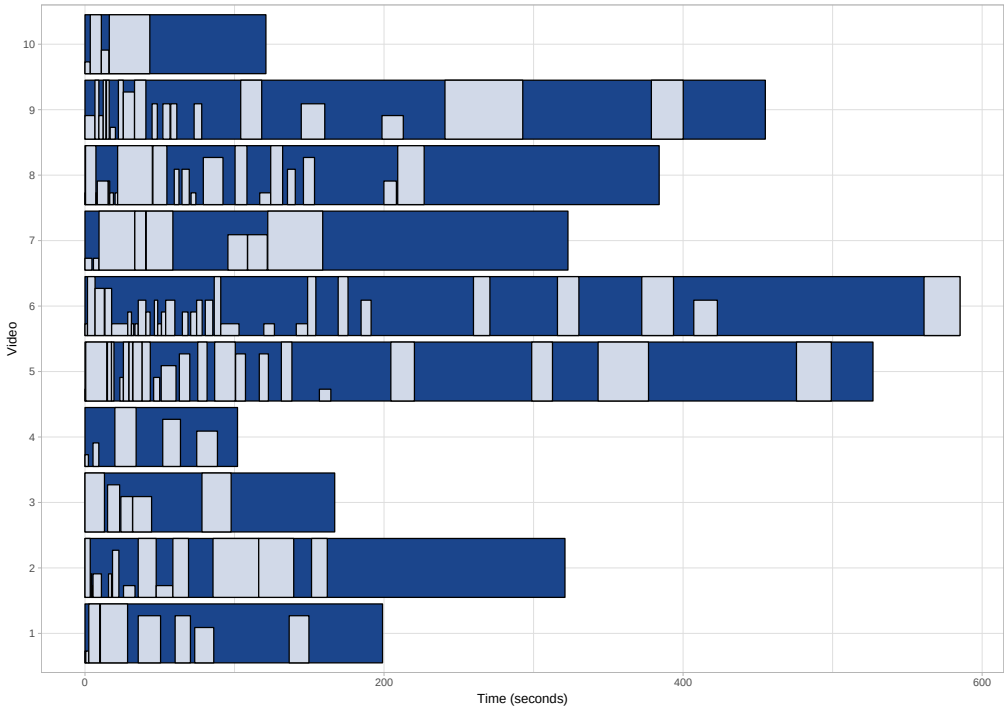

Fig. 5. Graphical representation of audio summarization performance

5 Subjective evaluation

In the previous section we had presented an evaluation methodology where each component is independently evaluated form the rest of the components. In this section we present an integral evaluation proposition where the system will be tested as a whole. This type of evaluation will involve the final users completing a questionnaire, which is considered to be the best indicator of quality for this type of system. There will be twelve participants who will be evaluating the summarized videos; six of these will be in Arabic (three men and three women per language). The participants must be at least 18 years old, with at least high school level education. If the participant has understanding problems and reading or writing impairment, they will not be allowed to participate in this evaluation.

Evaluation will be on various topics such as Politics, Soccer, War, Homosexuality and each participant will receive three videos. The questionnaires accurately analyze the quality of video summarization while taking into account the proposed resources. When the user first accesses the AMIS website, the evaluation will start with the user having to complete respondent information (see Figure 6) and some data about the human interfaces (see Figure 7) they usually use. Sociodemographic information on the evaluators will be obtained in order to enhance the last stage of data analytics. The

respondent will only be allowed to complete the text and video evaluation once these forms have been completed.

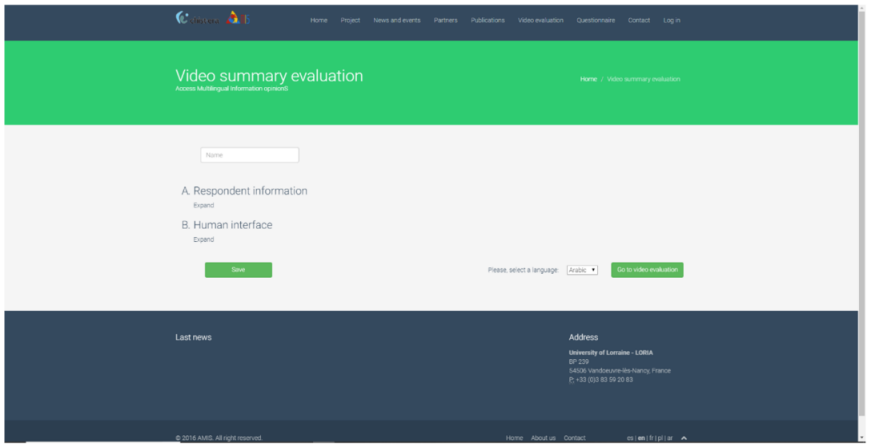

Fig. 6. Website evaluation Access

For each video, there will be two general questions. They will be on a scale from 0 (“Not done”) to 4 (“Excellent”). These answers will give an idea as to whether the summary is understandable or if any part do not make sense. Following this, depending on the video length, there will be 3-5 video specific questions which will have only three possible answers. The answers to the specific questions will allow us to see if the summarization has gathered the main ideas (see Figure 8).

6 Conclusion

This paper summarizes the implementation of the AMIS project with the production of a functional prototype allowing the capture of the main idea of a video in a foreign language (Arabic in this case). The result is a summary of the original video in English. This system provides good results but has to be improved obviously. This project was a real challenge since there are scientific locks for each used component: video summary, audio summary, text summary, automatic speech recognition and machine translation. Even the objective evaluation was a real problem since we do not have reference corpora to estimate the performance of the different components, but we overcame this issue by creating pseudo-reference corpora. A subjective evaluation was also conducted where several people evaluated the quality of the summaries.. Each component of AMIS corresponds to a research problem not completely solved, but we decided to tackle these different problems. Several lines of research have emerged and for which we proposed to find solutions. The serialization of components in the architecture of AMIS is a problem for which we are well aware, however the results obtained are not as bad as we

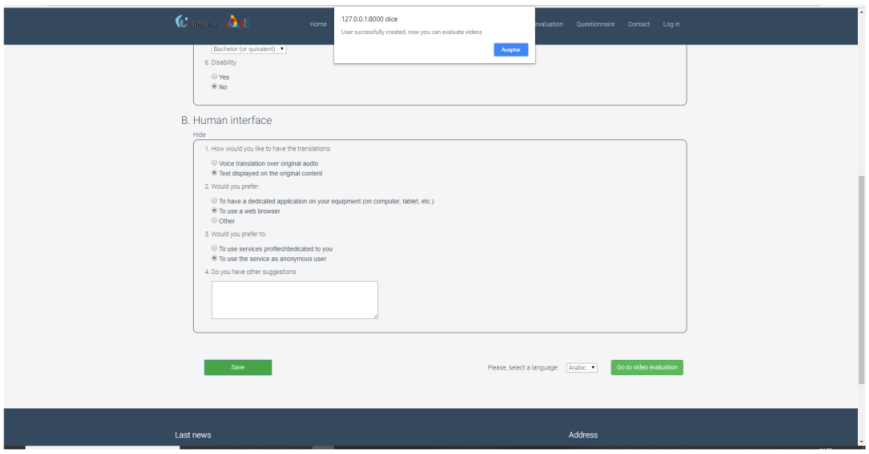

Fig. 7. Preliminary questionnaires

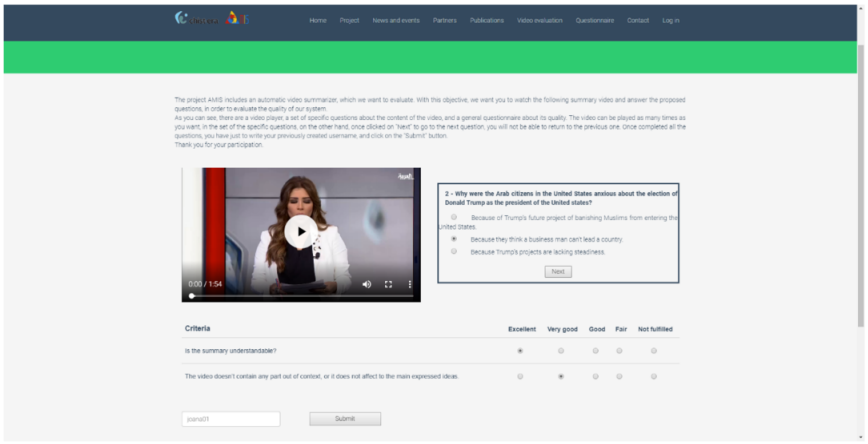

Fig. 8. Answering the questionnaire after video visualisation

imagined at the beginning. Indeed, the propagation of errors is a real handicap, but the inter-component collaboration has been made possible thanks to a well-thought-out arrangement of the different components. We are currently working on other types of architecture allowing more flexibility in communication to improve the results.

Acknowledgment

We want to acknowledge the support of Chist-Era for funding this work through the AMIS (Access Multilingual Information opinionS) project. Research work funded by

the National Science Center, Poland, conferred based on the decision number DEC-2015/16/Z/ST7/00559.

References

1. R. Baran and A. Zeja. The imcop system for data enrichment and content discovery and delivery. In *2015 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 143–146, Dec 2015.
2. Peter Bell, Catherine Lai, Clare Llewellyn, Alexandra Birch, and Mark Sinclair. A system for automatic broadcast news summarisation, geolocation and translation. In *INTERSPEECH*, pages 730–731, 2015.
3. Peter F Brown, Vincent J Della Pietra, Stephen A Della Pietra, and Robert L Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, 19(2):263–311, 1993.
4. Khalid Choukri, Mahtab Nikkhou, and Niklas Paulsson. Network of data centres (netdc): Bnsc-an arabic broadcast news speech corpus. In *LREC*, 2004.
5. Andreas Eisele and Yu Chen. Multiun: A multilingual corpus from united nation documents. In *LREC*, 2010.
6. Mark JF Gales. Maximum likelihood linear transformations for hmm-based speech recognition. *Computer speech & language*, 12(2):75–98, 1998.
7. Carlos-Emiliano González-Gallardo and Juan-Manuel Torres-Moreno. Sentence boundary detection for french with subword-level information vectors and convolutional neural networks. *arXiv preprint arXiv:1802.04559*, 2018.
8. M. Gygli, H. Grabner, and L. Van Gool. Video summarization by learning submodular mixtures of objectives. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3090–3098, June 2015.
9. M. Huang, A.B. Mahajan, and D.F. DeMenthon. *Automatic Performance Evaluation for Video Summarization*. AD-a448 064. Maryland Univ. College Park Inst. for Advanced Computer Studies, 2004.
10. Denis Jouvet, David Langlois, Mohamed Amine Menacer, Dominique Fohr, Odile Mella, and Kamel Smaïli. Adaptation of speech recognition vocabularies for improved transcription of youtube videos. In *ICNLSSP Conference*, 2017.
11. Muhammad Usman Ghani Khan, Rao Muhammad Adeel Nawab, and Yoshihiko Gotoh. Natural language descriptions of visual scenes corpus generation and analysis. In *Joint Workshop on Exploiting Synergies between Information Retrieval and Machine Translation (ESIRMT) and Hybrid Approaches to Machine Translation (HyTra)*, pages 38–47. ACL, 2012.
12. Reinhard Kneser and Hermann Ney. Improved backing-off for m-gram language modeling. In *icassp*, volume 1, page 181e4, 1995.
13. Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. Moses: Open source toolkit for statistical machine translation. In *45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. ACL, 2007.
14. Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. ACL, 2003.
15. Artur Komorowski, Lucjan Janowski, and Mikołaj Leszczuk. Evaluation of multimedia content summarization algorithms. In Kazimierz Choroś, Marek Kopel, Elżbieta Kukla, and Andrzej Siemiński, editors, *Multimedia and Network Information Systems*, pages 424–433, Cham, 2019. Springer International Publishing.

16. Mikołaj Leszczuk, Michał Grega, Arian Koźbiał, Jarosław Gliwski, Krzysztof Wasieczko, and Kamel Smaïli. Video summarization framework for newscasts and reports – work in progress. In Andrzej Dziech and Andrzej Czyżewski, editors, *Multimedia Communications, Services and Security*, pages 86–97, Cham, 2017. Springer International Publishing.
17. Yang Liu, Nitesh V Chawla, Mary P Harper, Elizabeth Shriberg, and Andreas Stolcke. A study in machine learning from imbalanced data for sentence boundary detection in speech. *Computer Speech & Language*, 20(4):468–494, 2006.
18. Annie Louis and Ani Nenkova. Automatically evaluating content selection in summarization without human models. In *2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 306–314. ACL, 2009.
19. Bente Maegaard, Khalid Choukri, Lise Damsgaard Jørgensen, and Steven Krauwer. Nemlar: Arabic language resources and tools. In *Arabic Language Resources and Tools Conference*, pages 42–54, 2004.
20. Inderjeet Mani. Summarization evaluation: An overview, 2001.
21. Inderjeet Mani and Mark T. Maybury. *Advances in Automatic Text Summarization*. MIT Press, Cambridge, MA, USA, 1999.
22. Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. In *14th python in science conference*, pages 18–25, 2015.
23. Mohamed Amine Menacer, Odile Mella, Dominique Fohr, Denis Jouvet, David Langlois, and Kamel Smaïli. Development of the Arabic Loria Automatic Speech Recognition system (ALASR) and its evaluation for Algerian dialect. In *ACLing 2017 - 3rd International Conference on Arabic Computational Linguistics*, pages 1–8, Dubai, United Arab Emirates, November 2017.
24. Paul Mermelstein. Distance measures for speech recognition, psychological and instrumental. *Pattern recognition and artificial intelligence*, 116:374–388, 1976.
25. Ani Nenkova. Automatic text summarization of newswire: Lessons learned from the document understanding conference. In *20th National Conference on Artificial Intelligence - Volume 3*, AAAI'05, pages 1436–1441. AAAI Press, 2005.
26. Franz Josef Och. Giza++: Training of statistical translation models. *http://www. isi. edu/˜ och/GIZA++. html*, 2001.
27. Franz Josef Och. Minimum error rate training in statistical machine translation. In *41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 160–167. ACL, 2003.
28. Karolina Owczarzak, John M. Conroy, Hoa Trang Dang, and Ani Nenkova. An assessment of the accuracy of automatic evaluation in summarization. In *Workshop on Evaluation Metrics and System Comparison for Automatic Summarization*, pages 1–9, Stroudsburg, PA, USA, 2012. ACL.
29. Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely. The kaldi speech recognition toolkit. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, 2011. IEEE Catalog No.: CFP11SRW-USB.
30. Alexandre Quemy, Krzysztof Jamrog, and Marcin Janiszewski. Unsupervised video semantic partitioning using ibm watson and topic modelling. In *Workshops of the EDBT/ICDT 2018 Joint Conference*, pages 44–49, March 2018.
31. Zafar Rafii and Bryan Pardo. Music/voice separation using the similarity matrix. In *ISMIR*, pages 583–588, 2012.
32. Aidean Sharghi, Jacob S. Laurel, and Boqing Gong. Query-focused video summarization: Dataset, evaluation, and a memory network based approach. In *2017 IEEE Conference on*

Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017, pages 2127–2136. IEEE Computer Society, 2017.
33. Andreas Stolcke. Entropy-based pruning of backoff language models. *arXiv preprint cs/0006025*, 2000.
34. Juan-Manuel Torres-Moreno. *Automatic Text Summarization*. Wiley and Sons, London, 2014.
35. Juan-Manuel Torres-Moreno, Horacio Saggion, Iria da Cunha, Eric SanJuan, and Patricia Velázquez-Morales. Summary evaluation with and without references. *Polibits*, 42:13–19, 2010.
36. Karel Veselỳ, Arnab Ghoshal, Lukás Burget, and Daniel Povey. Sequence-discriminative training of deep neural networks. In *Interspeech'13*, 2013.