Data Fusion based on Game Theory for Speaker Diarization

Marta Barrilero and Federico Alvarez
Grupo de Aplicación de Telecomunicaciones Visuales
Escuela Técnica Superior de Ingenieros de Telecomunicación (ETSIT)
Universidad Politécnica de Madrid, Spain
{mbg@gatv.ssr.upm.es, fag@gatv.ssr.upm.es}

Abstract—*A novel algorithm based on bimatrix game theory has been developed to improve the accuracy and reliability of a speaker diarization system. This algorithm fuses the output data of two open-source speaker diarization programs, LIUM and SHoUT, taking advantage of the best properties of each one. The performance of this new system has been tested by means of audio streams from several movies. From preliminary results on fragments of five movies, improvements of 63% in false alarms and missed speech mistakes have been achieved with respect to LIUM and SHoUT systems working alone. Moreover, we also improve in a 20% the number of recognized speakers, getting close to the real number of speakers in the audio stream.*

Keywords: speaker diarization, bimatrix game-theory, data fusion.

1. Introduction

1.1 Speaker diarization

Speaker diarization is the process of splitting an audio recording with an unknown number of speakers into time segments according to a recognized speaker, and then clustering those fragments by speakers. It is known as the task of determining *Who spoke when?* [1]. This technique distinguishes between instants of speech, music or silence. Initially, it emerged as an initial process for the speech recognition, but over recent years has become an important independent task used in many fields, such as information retrieval or navigation. In fact, it has been mainly developed for broadcast recorded meetings [2], news audio and telephone conversations [3]. Generally, speaker diarization is a very useful tool in every field in which it is necessary the determination of the number of speakers, along with the time periods when each one is active.

The most remarkable applications for speaker diarization systems are reflected in the Rich Transcription evaluation series (RT) [4], a community which promotes and gauges advances in the state-of-the-art in several automatic speech recognition technologies. This community is sponsored by the National Institute of Standards and Technology (NIST) in the Unites States [4].

Unfortunately, few speaker diarization systems are available for public general use. Two of them are LIUM [5] and SHoUT [6], which are open-source systems available on the Internet, mainly developed for applications on broadcast news that use audio streams. In this paper, both LIUM and SHoUT systems are used for obtaining speaker diarization output data on audio streams from movies. We use a fusion system to select the best decision at every second of our input audio from LIUM and SHoUT programs, in order to reach a better reliability and accuracy in our whole speaker diarization system. This fusion is carried out by means of a novel algorithm based on bimatrix game theory.

1.2 Game theory

Game theory is the study of mathematical techniques for analyzing situations of cooperation or conflict (games) between two or more rational intelligent individuals (players) [7]. It was introduced by von Neumann in 1928 [8], and improved later together with Oskar Morgenstern by considering cooperative games of several players [9]. Subsequently, Nash introduced the *Nash equilibrium* criterion [10], which supposed an important advance. Game theory has been widely used in a great variety of fields, such as economics, political science, philosophy or engineering. One of its main goals is the study of intelligent rational decision making, since it allows the understanding and modeling of different rational strategies performed by diverse agents. This involves a considerable help and support in decision making systems.

The three main elements in a game are the players (typically two), their played strategies, and the pay-off matrix, which contains the received prize by each player for each strategy. *Zero-sum games* [7] are a particular case in which the interest of both players are strictly opposed, hence the pay-off matrix contain the same values in opposite signs. Players in this kind of games strictly play in a competitive way. On the other hand, *non-zero-sum games* [11] can contain some strategies which are equally beneficial or detrimental for both players. Therefore, in *non-zero-sum games* competitive and cooperative strategies are generally performed.

The modeled game in this paper is *non-zero-sum*. It executes a data fusion algorithm based on the comparison of two data outputs from two systems. These data can be processed

in a cooperative way, taking into account the outcome from both data sources, or in a competitive way, analyzing the different properties of every system, and selecting the best of them.

Fusion data systems related to game theory have been widely studied, mainly in military and general security fields. For instance, a data fusion aided platform routing algorithm based on game theory for cooperative intelligence is proposed by Shen et al. [12]. Other example is using a Markov stochastic game method to estimate the belief of possible cyber attacks patterns [13]. Although similar algorithms to the proposed in this paper have not been found in the literature, game theory is a powerful tool that can be successfully used for improving speaker diarization by means of fusion data algorithms.

2. Speaker diarization systems

2.1 LIUM

LIUM [14] is an open toolkit [5] for speaker diarization developed by the *Laboratoire d'Informatique de l'Université du Maine* [15] for the French ESTER2 evaluation campaign [16]. It obtained the best results for the task of speaker diarization of broadcast news in 2008. It is written in Java and performs methods such as Mel-frequency cepstral coefficients computation, speech/non-speech detection and speaker diarization, using some tools from the speech recognition system Sphinx-4 [17]. The main algorithm follows next steps:

1) Segmentation based on BIC (Bayesian Information Criteria): speaker change points are detected and the audio signal is split into segments.
2) BIC clustering: similar segments are joined together.
3) Segmentation based on Viterbi decoding: a new segmentation is generated.
4) Speech detection: in order to remove music and jingle regions, a segmentation into speech / non-speech is obtained using a Viterbi decoding with 8 one-state Hidden Markov Model.
5) Gender and bandwidth detection: detection of gender and bandwidth is done using a Gaussian Mixture Model (GMM) for each of the 4 combinations of gender (male / female) and bandwidth (narrow/ wide band). Each cluster is labeled according to the characteristics of the GMM which maximizes likelihood over the features of the cluster.
6) GMM-based speaker clustering: a hierarchical agglomerative clustering is performed over the last diarization in order to obtain a one-to-one relationship between clusters and speakers.

Audio input files used in the executions of the system are in Wave format (16kHz / 16bit PCM mono). Output files show the properties of the analyzed segment, along with several frequency values related to each found cluster.

Figure 1 shows an example of output file in LIUM. It provides the instants in which every speaker change is recognized, the speech segment length in centi-seconds, the gender of the speaker and the speaker label.

```
;; cluster:S10 [ score:FS = -33.40578264060857 ] [ score:FT = -33.96415644658132 ] [
score:MS = -33.946822434659126 ] [ score:MT = -34.11794073282604 ]
cena 1 1409 201 F S U S10
cena 1 2969 286 F S U S10
cena 1 3313 516 F S U S10
cena 1 3917 405 F S U S10
cena 1 5123 451 F S U S10
cena 1 5617 227 F S U S10
cena 1 6980 304 F S U S10
cena 1 7847 186 F S U S10
cena 1 8033 421 F S U S10
cena 1 17514 424 F S U S10
cena 1 29032 636 F S U S10
cena 1 36641 1098 F S U S10
cena 1 41561 500 F S U S10
cena 1 43503 292 F S U S10
;; cluster:S11 [ score:FS = -33.30782570921103 ] [ score:FT = -33.48750761571984 ] [
score:MS = -32.94431915767501 ] [ score:MT = -33.2660012560795 ]
cena 1 4774 349 M S U S11
cena 1 5968 312 M S U S11
cena 1 6328 416 M S U S11
```

Fig. 1: LIUM output file

2.2 SHoUT

SHoUT is a Dutch acronym for *Speech Recognition Research at the University of Twente*. It was completely developed by Marijn Huijbregts during his PhD research [18] at the University of Twente. It is also open source [6], and it is written in C++ in a Linux platform. SHoUT program follows next steps:

1) Speech activity detection: a silence-based segmentation strategy is employed based on silence and sound models training using HMM.
2) Segmentation and clustering: a new segmentation is performed merging multiple models throughout several iteration steps.
3) Automatic speech recognition: several features are extracted and normalized by means of cepstrum mean normalization and vocal tract length normalization. These features are decoded in order to obtain acoustic models dependent on speech clusters.
4) Acoustic model adaptation: the clustering information obtained during segmentation and clustering is used to create speaker dependent acoustic models.

The output file from SHoUT is shown in Figure 2. It is similar to the LIUM one previously explained, although the information length is provided using seconds as time unit.

3. Game modeling

A data fusion system has been designed according to a bimatrix game with the three main elements in every game: players, strategies and pay off.

```
SPKR-INFO SpeechNonSpeech 1 <NA> <NA> <NA> unknown SPK01 <NA>
SPEAKER SpeechNonSpeech 1 0.000 0.740 <NA> <NA> SPK01 <NA>
SPEAKER SpeechNonSpeech 1 89.780 0.740 <NA> <NA> SPK01 <NA>
SPEAKER SpeechNonSpeech 1 91.280 6.070 <NA> <NA> SPK01 <NA>
SPEAKER SpeechNonSpeech 1 98.110 2.030 <NA> <NA> SPK01 <NA>
SPEAKER SpeechNonSpeech 1 100.900 0.830 <NA> <NA> SPK01 <NA>
SPEAKER SpeechNonSpeech 1 102.490 0.980 <NA> <NA> SPK01 <NA>
SPEAKER SpeechNonSpeech 1 104.750 3.720 <NA> <NA> SPK01 <NA>
SPEAKER SpeechNonSpeech 1 109.520 0.740 <NA> <NA> SPK01 <NA>
SPEAKER SpeechNonSpeech 1 111.600 0.810 <NA> <NA> SPK01 <NA>
SPKR-INFO SpeechNonSpeech 1 <NA> <NA> <NA> unknown SPK05 <NA>
SPEAKER SpeechNonSpeech 1 119.220 1.470 <NA> <NA> SPK05 <NA>
SPEAKER SpeechNonSpeech 1 131.500 1.030 <NA> <NA> SPK05 <NA>
SPKR-INFO SpeechNonSpeech 1 <NA> <NA> <NA> unknown SPK02 <NA>
SPEAKER SpeechNonSpeech 1 132.530 3.210 <NA> <NA> SPK02 <NA>
SPEAKER SpeechNonSpeech 1 138.300 1.050 <NA> <NA> SPK02 <NA>
SPEAKER SpeechNonSpeech 1 144.460 1.030 <NA> <NA> SPK02 <NA>
SPEAKER SpeechNonSpeech 1 145.490 0.710 <NA> <NA> SPK01 <NA>
SPEAKER SpeechNonSpeech 1 148.030 1.820 <NA> <NA> SPK01 <NA>
SPEAKER SpeechNonSpeech 1 150.610 0.950 <NA> <NA> SPK01 <NA>
SPEAKER SpeechNonSpeech 1 152.320 0.500 <NA> <NA> SPK01 <NA>
SPEAKER SpeechNonSpeech 1 152.820 0.340 <NA> <NA> SPK05 <NA>
SPEAKER SpeechNonSpeech 1 157.730 0.740 <NA> <NA> SPK05 <NA>
```

Fig. 2: SHoUT output file

A. *Players*

We consider two players: A (LIUM) and B (SHoUT), which perform their strategies throughout the whole audio sample. Each played strategy corresponds to one second. Thus, the duration of the audio sample will determine the number of times the game is played, which is the same as the total number of performed strategies by each player.

B. *Strategies*

Each player can carry out three different strategies:

- *Strategy nº 1: No-change speaker*
  There is not a speaker change detection. The detected speaker in the current second is the same as the detected in the previous one.

- *Strategy nº 2: New speaker*
  A speaker change to an unknown speaker is detected, since the detected speaker in the current second is different to the detected one in the previous second. In addition, the current speaker is new, since he/she has not been previously detected in the audio sample.

- *Strategy nº 3: Former speaker*
  A speaker change is detected in the same way as in the previous strategy. However, the detected speaker in the current second has already been detected previously in the audio sample.

C. *Pay-off*

Initially, the pay-off matrixes have the same structure as a typical bimatrix game with two players and three strategies.

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \; B = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} \tag{1}$$

The coefficients a_{ik} and b_{ik} stand for the pay-off received by player A (LIUM) and player B (SHoUT), respectively, if LIUM plays strategy i and SHoUT plays strategy k.

The fusion data system is designed according to a decider for this game. This decider performs the following steps. For each second:

1) Select LIUM and SHoUT strategy.
2) Compare both pay-offs, related to both strategies.
3) Choose the strategy which supposes highest pay-off for the corresponding player. The fusion system will adopt the selected strategy in current second.
4) Each player receives its pay-off.

Pay-off values have been decided according to the different observed behaviors in both diarization systems. The most reliable strategies have been assigned in line with the highest pay-off values. If a system is chosen by the decider, it receives a positive pay-off, whereas if the system has not been chosen, it receives the same pay-off value with negative sign. So far, a zero-sum game would be appropriate to model this behavior. However, there is another situation which must be taken into account: cases in which both systems perform the same strategy have been considered as the most reliable. In these cases, the decider certainly choose both systems. In addition, the selected strategy is provided with the maximum reliability since it is supported by two systems. Hence, these cases have been assigned to the highest pay-off values in both systems. Thus, a player receives a good pay-off if its strategy is chosen, but it receives an even better pay-off if both players are chosen. Consequently, the game is non-zero sum, and two pay-off matrixes are needed: therefore, the game is bimatrix. Moreover, several ideas that cause modifications in some pay-off values will be introduced subsequently.

To establish the required principles to set the pay-off values it is necessary to consider several properties related to both systems, which have been observed by means of several tests executed in both systems with different audio streams. The main goal of these considerations is to determine a direct relation between the reliability of each system and the pay-off values. On this way, the decider will select the most reliable strategy for each event, according to the pay-off values. The following properties have been implemented:

1) Initially, the pay-off related to different strategies for each player ($i \neq k$) are the same value with opposite

signs. Thus:

$$a_{ik} = -b_{ik} \qquad \forall i \neq k \tag{2}$$

2) If both systems execute the same strategy, the decider will certainly adopt it. Moreover, this strategy will be provided with the maximum reliability. Thus, in this case the pay-off values are the same and have the maximum value:

$$a_{ii} = b_{ii} \qquad \forall 1 \leq i \leq 3 \tag{3}$$
$$a_{ii} > a_{ik} \qquad \forall i \neq k \tag{4}$$
$$b_{ii} > b_{ik} \qquad \forall i \neq k \tag{5}$$

Note that these situations correspond to Nash equilibria [10], since they are maximum in both matrixes, i.e. they are optimum, and any other situation would suppose worse results. So, if both systems executed the same strategies all the time, the game would be constantly in an equilibrium point, reaching the maximum reliability.

3) Both systems perform silence, music and speech activity recognition with excellent results. In fact, the most remarkable found mistakes are related to errors in correct speaker identification. Specifically, both systems tend to consider former speakers as new speakers. Consequently, a detection of a former speaker (strategy nº3) is more reliable than a detection of a new speaker (strategy nº2). In addition, taking into account the negative pay-off values for non-chosen players, the final pay-off values fulfill the next properties:

$$a_{3i} > a_{21} > 0 \tag{6}$$
$$a_{i3} < a_{12} < 0 \tag{7}$$
$$b_{i3} > b_{12} > 0 \tag{8}$$
$$b_{3i} < b_{21} < 0 \tag{9}$$
$$\forall i < 3$$

Note that the pay-off values in entries below the main diagonal in A matrix are positive, since it supposes that the player A has been chosen. Same way in player B, but in this case the entries above the main diagonal are positive.

4) LIUM is more likely to detect erroneously new speakers than SHoUT. In fact, the final number of detected speakers by LIUM is commonly higher than the one by SHoUT. Therefore, a new speaker change detection in LIUM (strategy nº2) when SHoUT has not detected a change (strategy nº1) will be considered as less reliable than the opposite case:

$$a_{21} < b_{12} \tag{10}$$

4. Evaluations results

Several tests have been performed in order to evaluate the results of our fusion system. We have taken for the experiments audio segments with dialogues with different speakers from the movies *"Guess Who's Coming to Dinner"*, *"Marnie"*, *"Pride and Prejudice"*, *"Psycho"* and *"Sense and Sensibility"*. All the segments have been manually labeled to obtain the ground truth, creating a file with the same format as output files from LIUM and SHoUT programs. This reference file has the time slots from the audio sample associated to the proper speaker. To perform a numerical evaluation of the system, specific values have been assigned to A and B matrixes, according to the established properties on section 3.

$$A = \begin{pmatrix} 50 & -10 & -20 \\ 10 & 40 & -30 \\ 20 & 30 & 60 \end{pmatrix} \; B = \begin{pmatrix} 50 & 15 & 20 \\ -10 & 40 & 30 \\ -20 & -30 & 60 \end{pmatrix} \tag{11}$$

Therefore, numerical prizes have been settled to each player at the end of the execution of the fusion system. The prize values depend on the duration of each segment, since a different prize is obtained for each second. The obtained results are shown on Table 1, where it can be observed that SHoUT has *won the game* in every movie except in *"Psycho"*.

Table 1: Obtained prizes values.

Movie	LIUM	SHoUT	Duration (seconds)
Guess Who's Coming to Dinner	18190	18340	438
Marnie	23560	23620	603
Pride and Prejudice	19370	19555	463
Psycho	19500	19300	446
Sense and Sensibility	7120	7490	157

To evaluate the improvements introduced by our speaker diarization system, we have used the *Diarization Error Rate (DER)* value [20]. This parameter can be defined as the addition of the different possible errors of a speaker diarization system. In our experiment, *DER* has been defined as:

$$DER = FalseAlarmSpeech + MissedSpeech \tag{12}$$

False Alarm Speech is the number of times where a hypothesized speaker change is labeled as a non-change speaker in the reference. On the contrary, *Missed Speech* is the number of times where a hypothesized non-change speaker is labeled as a speaker change in the reference.

Silence activity detection can be also considered in a *DER* measure. Nevertheless, we have not taken it into account, since we have tested that the presented results by both systems in silence activity detection are much better than the results in speaker identification task, as it was explained

in section 3. Thus, only time slots related to detected speech have been considered.

The results obtained with LIUM, SHoUT, and our fusion system in terms of *DER*, are shown on Table 2, together with the added improvements (in %) by our system:

Table 2: *DER* and improvement values.

	DER			Improvements (%)	
Movie	LIUM	SHoUT	Fusion	LIUM	SHoUT
Guess Who's Coming to Dinner	22	19	4	82	79
Marnie	14	7	2	86	71
Pride and Prejudice	14	15	7	50	53
Psycho	12	11	3	75	73
Sense and Sensibility	5	10	4	20	60

On average, improvements of 63% and 67% on the *DER* parameter with respect to LIUM and SHoUT programs are reached with our fusion system. The less improvements correspond to those films, *"Pride and Prejudice"* and *"Sense and Sensibility"*, in which LIUM system performs lower *DER* than SHoUT. This can be translated into a possible unfairly performance of the fusion system, since the system which presents better results has *lost the game*. We are working to improve our system in these cases in order to reduce even more the *DER* parameter independently on the input stream.

In addition, the number of recognized speakers has been evaluated. As it has been previously explained, it is difficult to recognize properly former speakers, so the number of recognized speakers by speaker diarization systems is typically greater than the true value. However, our system reduces this number thanks to the fusion of strategies which allows clustering some speakers previously considered as different. The results are shown on Table 3. On average, we improve a 20% the number of detected speakers in comparison to the best output from LIUM and SHoUT systems working alone.

Table 3: Number of recognized speakers.

	Real speakers	Recognized			Improvements (%)	
Movie		LIUM	SHoUT	Fusion	LIUM	SHoUT
Guess Who's Coming to Dinner	4	19	8	6	68	25
Marnie	4	16	5	5	69	0
Pride and Prejudice	7	16	12	10	37	17
Psycho	5	8	9	6	25	33
Sense and Sensibility	3	6	7	4	33	43

5. Conclusions

In this paper we introduce a novel fusion algorithm based on game theory to improve the accuracy and reliability of two speaker diarization systems: LIUM and SHoUT. This algorithm is based on bimatrix game theory and performs a non-zero sum game. The establishment of the pay-off values has been carried out by means of different properties inferred by the analysis from both systems' behavior. The obtained tests show considerable advances of our fusion system in comparison to the results from LIUM and SHoUT programs working alone.

ACKNOWLEDGMENT

This paper is based on work performed in the framework of the Spanish national project BUSCAMEDIA (CEN-20091026), which is partially funded by the "CDTI-Ministerio de Economía y Competitividad".

References

[1] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, O. Vinyals. "Speaker Diarization: A Review of Recent Research," *IEEE Transactions on Audio, Speech, and Language Processing* vol. 20, no.2, pp. 356-370, Feb. 2012.

[2] X. Anguera, C. Wooters, B. Peskin, M. Aguilo, "Robust speaker segmentation for meetings: The ICSI-SRI Spring 2005 Diarization System," in *Proc. Machine Learning for Multimodal Interaction Workshop (MLMI)*, Edinburgh, U.K., pp. 402-414, 2005.

[3] C. Barras, X. Zhu, S. Meignier,J-L. Gauvain. "Multi-Stage Speaker Diarization of Broadcast News", *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1505-1512, Sep. 2006.

[4] "The NIST Rich Transcription 2009 (RT'09) Evaluation", NIST 2009. [Online]. Available: http://www.itl.nist.gov/iad/mig/tests/rt/

[5] LIUM website. [Online]. Available: http://lium3.univ-lemans.fr/diarization/

[6] SHoUT website. [Online]. Available: http://shout-toolkit.sourceforge.net/use_case_diarization.html

[7] R. B. Myerson. "Game theory: analysis of conflict". Harvard University Press, ISBN 978-0-674-34116-6, 1997.

[8] J. v. Neumann. "Zur Theorie der Gesellschaftsspiele", Mathematische Annalen, 100(1), pp. 295-320, 1928.

[9] O. Morgenstern, J. v. Neumann. "The Theory of Games and Economic Behavior", Princeton University Press, 1947.

[10] J. Nash. "Equilibrium points in n-person games", *Proc. of the National Academy of Sciences*, pp. 48-49, 1950.

[11] A. Scodel, J. Sayer Minas, P. Ratoosh, M. Lipetz. "Some Descriptive Aspects of Two-Person Non-Zero-Sum Games" . *The Journal of conflict resolution 3* (195 9), pp. 114-119, JSTOR, 1959.

[12] D. Shen, G. Chen, J. Cruz, E. Blasch, "A game Theoretic Data Fusion Aided Path Planning Approach for Cooperative UAV Control," *IEEE Aerospace Conf., Big Sky, MT*, Mar. 2008.

[13] D. Shen, G. Chen, E. Blasch, G. Tadda, "Adaptive Markov Game Theoretic Data Fusion Approach for Cyber Network Defense", *IEEE Military Communications Conference*, 2007.

[14] S. Meignier, T. Merlin. "Lium SpkDiarization: An open source toolkit for diarization". In *CMU SPUD Workshop*, Dallas, Texas, USA, 2010.

[15] Laboratoire d'Informatique de l'Université du Maine website. [Online]. Available: http://www-lium.univ-lemans.fr

[16] P. Deléglise, Y. Estève, S. Meignier, T. Merlin, "Improvements to the LIUM French ASR system based on CMU Sphinx: what helps to significantly reduce the word error rate?", in *Interspeech*, Sep. 2009.

[17] W. Walker, P. Lamere, P. Kwok, B. Raj, R. Singh, E. Gouvea, P. Wolf, J. Woelfel. "Sphinx-4: A flexible open source framework for speech recognition", Tech. Rep., Sun Microsystems Inc., 2004.

[18] M. Huijbregts. "Segmentation, Diarization, and Speech Transcription: Surprise Data Unraveled". *PhD Thesis*, University of Twente, The Netherlands, 2008.

[19] C. Wooters, C., M. Huijbregts. "The ICSI RT07s speaker diarization system", *Multimodal Technologies for Perception of Humans. Lecture Notes in Computer Science*, 2007.