Expert Systems With Applications 213 (2023) 119238

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Speech and multilingual natural language framework for speaker change detection and diarization

Or Haim Anidjar [a,b,d,*], Yannick Estève [c], Chen Hajaj [d,e], Amit Dvir [a,b], Itshak Lapidot [f,c]

[a] Ariel Cyber Innovation Center, Ariel University, Golan Heights 1, 4077625, Ariel, Israel
[b] Department of Computer Science, Ariel University, Golan Heights 1, 4077625, Ariel, Israel
[c] Avignon University, Laboratoire informatique d'Avignon, 339, chemin de Meinajariš BP 9122884 911 Avignon cedex 9, France
[d] Data Science and Artificial Intelligence Research Center, Ariel University, Golan Heights 1, 4077625, Ariel, Israel
[e] Department of Industrial Engineering & Management, Ariel University, Golan Heights 1, 4077625, Ariel, Israel
[f] Afeka, Tel-Aviv Academic College of Engineering, Afeka Center of Language Processing, 38 Mivtza Kadesh St, Tel-Aviv, 6998812, Israel

ARTICLE INFO

Keywords:
Speaker change detection
Transformers
Speech recognition
Speaker diarization
Speaker embedding

ABSTRACT

Speaker Change Detection (SCD) is the problem of splitting an audio-recording by its speaker-turns. Many real-world problems, such as the Speaker Diarization (SD) or automatic speech transcription, are influenced by the quality of the speaker-turns estimation. Previous works have already shown that auxiliary textual information (for mono-lingual systems) can be of great use for detection of speaker-turns and the diarization systems' performance. In this paper, we suggest a framework for speaker-turn estimation, as well as the determination of clustered speaker identities to the SD system, and examine our approach over a multi-lingual dataset that consists of three mono-lingual datasets—in English, French, and Hebrew. As such, we propose a generic and language-independent framework for the SCD problem that is learned through textual information using state-of-the-art transformer-based techniques and speech-embedding modules. Comprehensive experimental evaluation shows that (i) our multi-lingual SCD framework is competitive enough when compared to a framework over mono-lingual datasets, and that (ii) textual information improves the solution's quality compared to the speech signal-based approach. In addition, we show that our multi-lingual SCD approach does not harm the performance of SD systems.

1. Introduction

Speaker Change Detection (SCD) is a well-known and central component in many applications. For example, in automatic speech transcription, or indexing of audio (Liu & Kubala, 1999; Lu & Zhang, 2002), SCD serves as a core component, since adaptation of speakers is highly beneficial. Another application in which SCD is central is the problem of Speaker Diarization (SD) (Anidjar, Lapidot, Hajaj and Dvir, 2021; Wan, Wang, Papir, & Moreno, 2018; Wang, Downey, Wan, Mansfield, & Moreno, 2018; Zhang, Wang, Zhu, Paisley, and Wang, 2019). In speech recognition, an SD system tries to infer "who is speaking and when?" in an audio-recording, by computing the time intervals of any speech utterance, and a clustered-label of speaker identity. Thus, the SCD problem is the heart of these kinds of systems, since given that the segmentation process is poor, no diarization or audio-indexing system can perform in an acceptable manner.

Basically, an SD system is composed of the following components: (i) a Voice Activity Detection (VAD) engine, which locates and omits non-speech segments, and next split the audio-recording into speech utterances that are supposed to be spoken by one speaker each, i.e., SCD (India Massana, Rodríguez Fonollosa, & Hernando Pericás, 2017; Luo, Wu, Zheng, & Wang, 2010; Meng, Mou, & Jin, 2017; Park & Georgiou, 2018; Zajíc, Soutner, Hrúz, Müller, & Radová, 2018); (ii) a module of embedding extraction, on which speaker-discriminative embeddings such as the Mel Frequency Cepstral Coefficients (MFCC) (Sunitha & Chandra, 2015); (iii) a component that estimates the number of speakers in the audio recording, and (iv) a clustering module that clusters speaker identities to the speech utterances (in case of unsupervised version of the SD problem).

Even though both vocal and textual information are informative for the SCD problem, only a few efforts to find a hybrid solution that combines speech and text analysis were developed for the SCD problem. The model in Park and Georgiou (2018) has used the one-hot-encoding approach as textual features, along with MFCC representation as vocal features, for a solution to the SCD problem. In addition, a new loss

* Corresponding author at: Department of Computer Science, Ariel University, Golan Heights 1, 4077625, Ariel, Israel.
E-mail addresses: orchaimanidjar@gmail.com (O.H. Anidjar), yannick.esteve@univ-avignon.fr (Y. Estève), chenha@ariel.ac.il (C. Hajaj), amitdv@ariel.ac.il (A. Dvir), itshakl@afeka.ac.il (I. Lapidot).

https://doi.org/10.1016/j.eswa.2022.119238
Received 2 August 2022; Received in revised form 20 October 2022; Accepted 5 November 2022
Available online 11 November 2022

function was proposed for selecting both the speaker-turns, and the most likely speaker, in any speech utterance. Then, it has been demonstrated that textual information provides auxiliary information to the vocal modality. The conclusions from the experimental evaluation have shown that a model that was trained on these two modalities improves the speaker diarization results—compared to a model that only exploits one of these two modalities at a single time.

Generally, speech-recognition solutions are considered language-agnostic for problems like SCD or SD. Yet, aggregation of textual information produces higher quality for this kind of problems (Anidjar, Lapidot, Hajaj, Dvir, 2021; Anidjar, Lapidot, Hajaj, Dvir and Gilad, 2021; Park & Georgiou, 2018; Park, Han, Lane, & Georgiou, 2018) - and thus SD systems are no longer considered as language agnostic, and are beneficialized from textual analysis, which may differ from one language into another. Clearly, more than one language can appear in an audio-recordings dataset, yet the assumption is that each audio-recording is mono-lingual (in the scope of this work), hence it is important that the textual SCD module will be able to cope with a multilingual dataset; the basic-version for the language model on which the text-encoding method in this work is based contains about 350 million parameters. Therefore, handling multiple language models is computationally expensive in terms of resource allocation, memory, runtime, and even applicability in a real-time environment. As a result, a mechanism that handles multiple languages more efficiently and feasibly is required.

As such, in this paper, a hybrid feature encoding process is suggested as part of the language-free SCD framework. That is, the added value of our approach is manifested by exploiting both word and speaker embeddings. In addition to the vocal features and the textual ones, meta-data features about word duration and speech rate will be exploited. It is important to mention that one of the main contributions of this work is the adaptation and adjustments of many complex components, into an end-to-end system that solves the multilingual version of the SCD problem, and its application over real-life SD systems.

This paper suggests a language-independent speaker segmentation module for the SCD problem, which is capable of accurately detecting the points on which the identity of a speaker changes, without being dependent on the spoken language—but in a pre-trained word or sentence embedding module (Feng, Yang, Cer, Arivazhagan, & Wang, 2020; Sun et al., 2020). This independence matters as in some scenarios, for instance, in Interpol transcriptional investigations (Fobbe, 2022) in which different speakers can speak different languages, due to the presence of vocal numbers, interpreters, different mother-tongue of witnesses, etc.

In the scope of this work, as aforementioned, each audio-recording is mono-lingual. Thus, we assume a perfect language-identification over the dataset, and emphasize the contribution of our approach. This assumption is reasonable and admissible, as one of the main scenarios on which this assumption holds is the world of organizational Call-Centers (Deschamps-Berger, 2021; Firc & Malinka, 2022; Litvinov, 2021). The state of Israel, for instance, is a good example of the demand for multi-lingual support of call centers; Hebrew, Arabic, English, Russian, and Amharic—are just among the variety of spoken languages in Israel, and call centers are required to support these languages. As such, a typical call center would require input from the user (caller) which indicates its native language, which makes the need for a language identification module unnecessary. However, supporting multiple SCD and SD models for each such spoken language is computationally inefficient, and requires an extremely high amount of computing resources. Thus, the multilingual approach for the SCD and SD problems presented in this paper is of high utility, provided that it maintains the robustness and performance as if a model would be constructed for each language separately. We can summarize the contribution of this paper as follows:

- Cross-Lingual Vectors are suggested as the text-encoding method, in order to manipulate three datasets at once, i.e. to train one single model for the hybrid SCD model. Moreover, this framework can extremely easily be extended to more than 100 languages, due to its training procedure.
- A competitive SCD component for three languages—Hebrew, French and English, compared to monolingual and bilingual models.
- This work is an advanced future-work that was suggested in Anidjar, Lapidot, Hajaj, Dvir et al. (2021), in which it was already shown that textual information is of great utility for problems like SCD and SD which are vocal-based from nature. In this work, we reprove that this conclusion still holds, despite the multilingual datasets, and the manner that our approach copes with them.

Both SD and SCD models based solely on voice analysis should not be sensitive to a variety of languages. However, our approach for the SCD problems which is based on textual information as well—shows significant resilience with respect to solutions based solely on voice characteristics. This, of course, places a limit on the need for multilingual textual analysis, but Cross-Lingual Vectors (CLV) (Brychcín, 2020; Camacho-Collados et al., 2020; Fu et al., 2020; Ruder, Vulić, & Søgaard, 2019) is exactly the approach for addressing this multilingual analysis constraint. In the world of NLP, the concept of CLV aims to represent the meaning of words, as well as knowledge transfer across different languages. For instance, translation of the phrase *'I love you.'* from English to Hebrew is non-trivial, and requires a deep context analysis; as there are two options to translate the verb *love* into a feminine or masculine entity, and the object *you* has four translation options - {plural, single} × {feminine, masculine}, so that there are 8 translation options in total. Thus, such methods learn word-representations in joint embedding space, by one of the following four common approaches - (i) Monolingual Mapping; (ii) Pseudo-Cross-Lingual; (iii) Cross-Lingual Training; and (iv) Joint Optimization. Clearly, each of these four approaches has its own pros and cons.

The remainder of this paper is structured as follows: Section 2 surveys related work on the SCD and SD problems; Section 3 describes the SCD and SD models employed in this work, including the text-encoding method which we use; Section 4 describes the three mono-lingual datasets in this paper; Section 5 explains the evaluation metrics that are computed over the models in this work, and how to compute them; Section 6 examines experimental results, and Section 7 concludes and summarizes this paper. For ease of reading, we provide a list of abbreviations in Table 1.

2. Related work

Typically, SD (Fujita et al., 2019; Landini, Profant, Diez, & Burget, 2022; Silnova, Brümmer, Rohdin, Stafylakis, & Burget, 2020) systems are addressed in two steps (speaker-turn estimation and clustering for speaker identities). As for the speaker-turn estimation, which is the main aspect of this paper, the most common and basic method to tackle the SCD is one in which the speaker-turns are computed through a VAD engine, which sometimes might be followed by a Bayesian Information Criterion (BIC) (Chen, Lee, He, & Soong, 2020). More sophisticated methods are presented in (India Massana et al., 2017) that presented a speaker-turn module that is based on Long-Short Term-Memory (LSTM) (Shahid, Zameer, Mehmood, & Raja, 2020; Zhang, Ouyang, Zhang, Xue and Zheng, 2019) neural network, that exploits both vocal and textual information as in Luo et al. (2010) which developed a similar module, that is based on an algorithm for speaker recognition, in real-time applications environment. Additional textual approaches that tackled the SCD problem are presented in Zajíc et al. (2018), which designed an architecture of feeding an LSTM neural network, that computes the encoded vectors as one-hot encoding. Such a method is not feasible for a large data set, as the datasets presented

Table 1
List of abbreviations.

Abbreviation	Meaning
AHC	Agglomerative Hierarchical Clustering
AMS	Additive Margin Softmax
ANN	Artificial Neural Network
BERT	Bidirectional Encoder Representations from Transformers
BIC	Bayesian Information Criterion
CCA	Canonical Correlation Analysis
CLV	Cross Lingual Vectors
DER	Diarization Error Rate
EER	Equal Error Rate
GE2E	Generalized End-To-End
GRU	Gated Recurrent Unit
HDM	Hidden Distortion Model
HMM	Hidden Markov Model
LSTM	Long Short-Term Memory
MFCC	Mel Frequency Cepstral Coefficients
MLM	Masked Language Model
NLP	Natural Language Processing
PLDA	Probabilistic Linear Discriminant Analysis
RNN	Recurrent Neural Network
SCD	Speaker Change Detection
SD	Speaker Diarization
SW	Sliding Window
S2T	Speech-2-Text
TLM	Translation Language Model
TSMD	Text, Speech and Meta-Data
VAD	Voice Activity Detection
WER	Word Error Rate

in this work contain over 100,000 different words—which can lead to extreme resource exploitation. Another example of a hybrid approach to SCD and SD problems was presented in Park and Georgiou (2018), which constructed a Sequence-to-Sequence (Gehring, Auli, Grangier, Yarats, & Dauphin, 2017; Sutskever, Vinyals, & Le, 2014) mechanism, that is trained on one-hot-encoding and MFCC vectors (Fang et al., 2019).

In the world of Neural Networks, the Sequence-to-Sequence architecture is a unique class of Recurrent Neural Network (RNN) (Shahid et al., 2020), that are typically used (yet, not restricted) to solve intricate language problems, such as machine translation (Ortega, Mamani, & Cho, 2020), image captioning (Pan, Yao, Li, & Mei, 2020), conversational models (Tian et al., 2017), and text summarization (El-Kassas, Salama, Rafea, & Mohamed, 2021). This architecture transforms a sequence into another by use of an RNN, or more often LSTM or Gated Recurrent Unit (GRU) (Jin et al., 2020; Yang, Zuo, & Cui, 2019) to avoid the vanishing gradient problem. The main components are an encoder neural network, and a decoder one. The encoder transforms the input sequence into a hidden vector, and the decoder inverses this procedure, by turning the hidden vector into an output sequence, using previous outputs as input context of previous computations. The problem with these architectures (i.e., RNN, LSTM, GRU) is that they become inefficient whenever the interval between relevant information and its corresponding context turns larger. That is, the probability of preserving a context of a specific word that is far away from a current and vital word to that context, decreases exponentially with the distance. Thus, the attention mechanism (Firat, Cho, & Bengio, 2016; Li, Liu, Zhang, & Cheng, 2020; Wang, Zhang, Kan, Shan, & Chen, 2020) is used in a neural network in order to address the memory gap. Instead of encoding a sequence in a single hidden state, each word is encoded in a corresponding hidden state that is moved through decoding stage. Then, transformers (Katharopoulos, Vyas, Pappas, & Fleuret, 2020; Liu, Duh, Liu, & Gao, 2020) are served in order to solve the parallelization problem, such as Bidirectional Encoder Representations from Transformers (BERT) (Nozza, Bianchi, & Hovy, 2020; Xin, Nogueira, Yu, & Lin, 2020), or its multi-lingual version a.k.a the Language Agnostic-BERT (Feng et al., 2020; Sun et al., 2020), are used.

The second main component of SD systems, as argued earlier in this section, is speaker segments clustering. A pre-processing procedure that precedes a clustering algorithm over speech utterances, is speaker-embedding extraction (Le & Odobez, 2019; Silnova et al., 2020), in order to encode speech utterances. There are many different DNN architectures and different input features that are used for embeddings design such as log-mel-filterbank energies as in Kang, Roy, and Chow (2020) and Wan et al. (2018) and MFCC features as in Snyder et al. (2018) and Snyder, Garcia-Romero, Sell, Povey and Khudanpur (2018). However, it is important to note that overlapping speech (Galibert, 2013; Kunešová, Hrúz, Zajíc, & Radová, 2019) poses a limitation over the efficiency of typical diarization systems; the speaker-turns are complex to define (speech overlapping for instance) and sometimes difficult to locate (silent speech parts), and clustering algorithms would contain speaker mixture. In these cases, end-to-end neural diarization (Fujita et al., 2019; Fujita, Watanabe, Horiguchi, Xue, & Nagamatsu, 2020) system are of great interest, as the SD problem is considered and examined as it would be a multi-label classification problem. That is, Fujita et al. (2019, 2020) proposed a permutation-free objective function that minimizes the Diarization Error Rate (DER) (Deléglise, Esteve, Meignier, & Merlin, 2009). In this manner, this method can cope with overlapping speech utterances, by feeding to the neural network multi-speaker audio-recordings, along with the speech utterances labels.

2.1. Cross-Lingual vectors

Nowadays, SCD components are used for state-of-the-art SD systems, mainly those that exploit textual information, which are developed using the monolingual assumption. To the best of our knowledge, no work that tackled the SD or SCD problems has presumed the existence of a multiple-language environment. However, the exploitation of Cross-Lingual Vectors (Brychcín, 2020; Camacho-Collados et al., 2020; Fu et al., 2020; Ruder et al., 2019) is certainly a possible approach for tackling the SCD problem from the multiple-language aspect. In the world of Multiple-Language Modeling (Juan, Ismail, Ujir, & Hipiny, 2020; Yang & Xiang, 2019), the goal is to learn a shared embedding space (Barry, Boschee, Freedman, & Miller, 2020; Wu, Li, Hsieh, & Sharpnack, 2019) between words in *all* languages, a.k.a Cross-Lingual Vectors (CLV). Equipped with such a vectorial space, one is able to train a model on datasets in any pre-trained language. By projecting examples available in one language into this space, a model can simultaneously obtain the capability to perform predictions in all other languages.

Cross-lingual word-embeddings are beneficial for the following reasons: first, they allow to contrast of words meaning across different languages, which is vital for the induction of bilingual lexicons, or cross-lingual-based information retrieval. Second, cross-lingual word-embeddings allow NLP models to transmit between different languages, especially resource-rich (as English) and resource-poor (Hebrew) languages, due to the shared vectorial representation in a linear space. In recent years, various models that cope with learning cross-lingual representations have been introduced. Among such types of cross-lingual models, one can find two common CLV approaches:

(1). Monolingual Mapping - (Artetxe, Labaka, & Agirre, 2016) - which consists of training a set of monolingual word-embedding models, using a set of monolingual corpora. Next, the shared embedding space of this set of models is learned using a linear mapping between monolingual representations in different languages, as in the Canonical Correlation Analysis (CCA) (Bhowmik, Tripura, Hazra, & Pakrashi, 2020; von Lühmann, Li, Müller, Boas, & Yücel, 2020) algorithm.

(2). Cross-Lingual-Training - (Feng et al., 2020; Sun et al., 2020), models are trained over word embeddings representation on a parallel corpus. Then a cross-lingual constraint is applied between embeddings of different languages, in order to force a relatively close spatial positioning of similar words in different languages to be close to each other in a shared vector space.

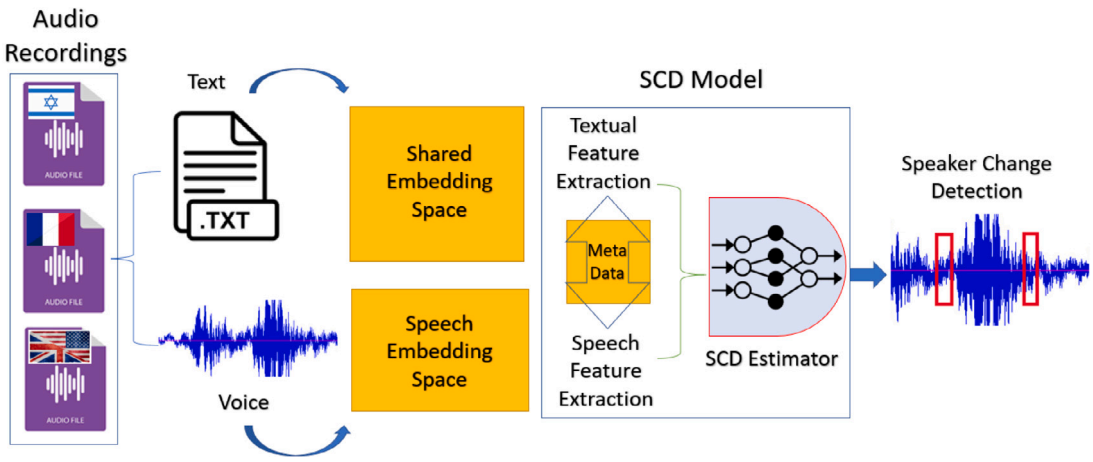

Fig. 1. An illustration of the suggested framework in this paper. For any audio-recording in a specific language (Hebrew, French, English), the automatic transcription as well as the spectrogram are extracted. Then, the textual information and the speech signal are extracted into features, along with meta-data features, by using the Shared-Embedding (Language Agnostic-BERT) and the Speech-Embedding components, which eventually feed a neural network that estimates speaker-turns. Finally, a diarization model is applied to the speech segments. Note that this figure is based on Anidjar, Lapidot, Hajaj, Dvir et al. (2021), and the aim of this paper is not an extension of Anidjar, Lapidot, Hajaj, Dvir et al. (2021) but a different and multilingual text-encoding method, and combined process of system's training.

3. System description

This work presents a multilingual solution to the SCD and SD problems, and exploits several components in order to compare its performance under different constraints and conditions; Section 3.1 presents the text-encoding method employed in this work, namely the Language Agnostic-BERT model (Feng et al., 2020) and its advantages due to its transformer-based architecture that is trained over a language model that supports the word order as part of its text-embedding process. Following that, Section 3.2 presents the voice-embedding computation module employed in this work. Next, Section 3.3 presents the SCD component and its model details, where the text-encoding method is replaced with the Language Agnostic-BERT one, and the whole SCD model's architecture is derived from the work presented by Anidjar, Lapidot, Hajaj, Dvir et al. (2021). Note that this figure is based on Anidjar, Lapidot, Hajaj, Dvir et al. (2021), and the aim of this paper is not an extension of Anidjar, Lapidot, Hajaj, Dvir et al. (2021) but a different and multilingual text-encoding method, and combined process of system's training. Finally, Section 3.4 presents the SD model on which the experimental evaluation process is tested, derived from the SD system that was presented by Park and Georgiou (2018) and Park et al. (2019). An illustration of the framework suggested in this paper can be seen in Fig. 1. It is important to note that the Audio-Recordings are replicated three times by three flags (for Hebrew, French, and English) in order to demonstrate that no matter what the spoken language is—the algorithmic flow remains the same for the entire framework.

3.1. Text-encoding Method - Agnostic BERT

The work in Anidjar, Lapidot, Hajaj, Dvir et al. (2021) tackles the SCD problem by using the pre-trained word embedding model published by Facebook corpora[1] for a dataset in the Hebrew language, which represents each word using a 300-dimensional vector. As the vectorial average encoding remains the same for word permutation in a given sub-sentence, and at the same time forces the utilization of multiple models for each language, we introduce a multilingual embedding technique. This technique is a mighty module that enables encoding text in different languages into a single shared word-embedding linear space, which can be applied over various NLP challenges (e.g., sentiment analysis, and text classification). In the scope of this work, we chose Language Agnostic-BERT (Feng et al., 2020) as the text-encoding method employed, since the Language Agnostic-BERT supports multiple-languages word and sentence embedding due to the utilization of transformers and the attention mechanism. We first describe the main idea behind Language Agnostic-BERT, and then Section 3.3 presents an explanation of the utilization of the Language Agnostic-BERT model in this work.

Language Agnostic-BERT - is an adaptation of the multilingual BERT (Devlin, Chang, Lee, & Toutanova, 2018) version that was adapted by Feng et al. (2020) in order to create language-agnostic sentence embeddings for different 109 languages (including English, French, and Hebrew). To do so, Masked Language Model (MLM) (Conneau & Lample, 2019) and Translation Language Model (TLM) (Conneau & Lample, 2019) are pre-trained, and examined through the ranking of language translation task, by using bi-directional as well as dual encoders. In this manner, the MLM and TLM have succeeded to create multilingual word and sentence embeddings, that eventually improved the average bi-text retrieval accuracy for these 109 languages, to 83.7%. As the dual encoder is able to produce word-embedding for any word in one of the 109 languages, the exploitation of the Language Agnostic-BERT architecture enables us to produce a dictionary of word embeddings for each word in the Hebrew, French and English datasets—languages that will be under examination in Section 6.

The Language Agnostic-BERT model is trained on a huge monolingual sentences corpus of size 17 billion, and a bilingual sentence pairs corpus of size 6 billion, by using the pre-trained MLM and TLM modules. It is important to note that there is no separate training process for any subset of the languages introduced in this work (Hebrew, French, English). In addition, the same Language Agnostic-BERT model[2] is being used throughout all the experimental evaluation process (Section 6), where the bidirectional dual encoder is trained using the Additive Margin Softmax (AMS) (Feng et al., 2020) loss function, as defined by Eq. (1):

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\frac{e^{\phi(x_i,y_i)-m}}{e^{\phi(x_i,y_i)-m}+\sum_{n=1,n\neq i}^{N}e^{\phi(x_i,y_n)}} \tag{1}$$

where $\phi(x,y)$ is the embedding space similarity of the sentence x and the sentence y, and defined as $\phi(x,y) = cosine(x,y)$ (see Eq. (3) for definition of $cosine(x,y)$). The goal of the AMS loss is to rank y_i over all

[1] Visit https://fasttext.cc/docs/en/crawl-vectors.html.

[2] Visit https://tfhub.dev/google/LaBSE/1.

the $N-1$ options in a specific batch of size N, where y_i is the translation of x_i, and $\phi(x_i, y_i)$ is reduced by m, which is the margin. Since $\mathcal{L}$ is asymmetric, and since the ranking of the AMS depends on whether y_i is the translation of x_i and vice versa, i.e. in another batch, it holds that x_i is the translation of y_i. The final loss should be symmetric and is given by Eq. (2):

$$\overline{\mathcal{L}} = \mathcal{L} + \mathcal{L}' \tag{2}$$

where for the bi-directional ranking, $\mathcal{L}'$ is defined equivalently as $\mathcal{L}$ such that the final loss function sums the source to target (pairs of x_i and y_i), $\mathcal{L}$, and target to source (pairs of y_i and x_i), $\mathcal{L}'$.

Note that the exploitation of the Language Agnostic-BERT is nontrivial. Unlike the typical BERT model, Language Agnostic-BERT contains a translation task as part of the training process, followed by a TLM. Moreover, the Language Agnostic-BERT is trained by a Transformer that produces cross-lingual sentence embeddings, which makes the embedding space similarity training harder. However, the dual encoders are trained through a translation ranking loss, in order to maximize the similarity of translated data-points pairs in the Shared Embedding Space. As a result, the Language Agnostic-BERT is capable of representing over 100 languages in one single model.

3.2. Voice-encoding method - Embeddings computation

The model presented by Fini and Brutti (2020) and Zhang, Wang et al. (2019) has exploited the speech embedder that was proposed by Wan et al. (2018), by using input utterances in order to learn speaker-discriminative embeddings. For this purpose, Wan et al. (2018) proposed the Generalized End-to-End (GE2E) loss function, whose training process is done by a parallel process of a respectable amount of speech segments at once, where each such batch contains a mixture of N different speakers, for which M speech segments of each speaker. According to the implementation[3] for the Speaker Verification module that was proposed by Wan et al. (2018), a text-independent speech-embedding neural network is trained over of any speaker and its speech utterances in AMI Corpus (Carletta et al., 2005). In our case, the embedding size is 256.

Observation. For any experiment in Section 6 that contains a training process of an SCD component, a *re-computation* of the speech embeddings for speaker segments is performed; as in the training process, the speech embeddings are computed over speech segments that consist of 3 words.

An SCD component over a test-file audio-recording returns all the locations on which a speaker-turn has been detected. Thus, in order to feed a clustering algorithm with an appropriate representation of the speech segments that have been detected using the SCD component, this speech-embedding computation is necessary.

3.3. Speaker change detection

The proposed approach to tackle the SCD problem takes advantage of the textual information, and the speech signal. Moreover, these two aspects produce meta-data features that are also beneficial. Without loss of generality, a generic data iterator is required, due to the tabular structure of the Speech-2-Text (S2T) based datasets, composed of text and speech features. As such, we adopt both the Sliding-Window (SW) (Ma et al., 2020) method and the neural network architecture from Anidjar, Lapidot, Hajaj, Dvir et al. (2021), as follows:

Text. In this work, we use a transformer-based technique, i.e. the Language Agnostic-BERT model (Feng et al., 2020) (Section 3.1), such that each sliding window is encoded from the textual standpoint with 1536 features; 768 for the first three words in a given SW, and another 768 for the last three words.

[3] Visit https://github.com/HarryVolek/PyTorch_Speaker_Verification.

Speech. Similarly to the Text component, each sliding window is encoded with a vector that consists of 512 features, so that 256 features represent the pronunciation of the first three words and another 256 for the last three words. Note that this computation is feasible, as the datasets in this work are based on a S2T engine, that provides the starting and ending time of each word in the audio-recording.

Each sub-vector of size 256 is a speaker embedding, and its computations are done by using the TI-SV speech-embedding module that was presented in Wan et al. (2018). This RNN-based architecture of the speech-embedding module enables getting a speech-embedding vector of fixed size (256 features), for speech segments of different lengths.

Finally, the speech embedding vector of size 512 is concatenated for each sliding-window, to the existing 1536 from the text-encoding process. In Section 3.2 we presented a deep explanation that refers to the embeddings computation, which uses a VAD engine. Thus, any of the experiments in Section 6 relies on a VAD engine as part of it.

Meta-Data. The final component that is added to the representation vector for each SW, consists of meta-data features. That is, both from the textual information and the speech signal, 14 features are extracted and concatenated for every sliding-window (that contains 6 words in it) as follows;

- The Duration of any of the 6 words (in seconds).
- Speech rate of the 6 words—calculated as the characters amount of any word, divided by the word's duration.
- Time elapsed between the 3rd and 4th words—might imply about a speaker-turn, if the delay is long enough.
- Euclidean distance between the two sub-vectors (speech embeddings), each of size 256, which might imply about a speaker-turn provided that the distance is far enough.

Finally, the input layer size of the neural network used for the SCD component (derived from the architecture in Anidjar, Lapidot, Hajaj, Dvir et al. (2021)) model is 2062: for the text component 1536 features, another 512 for the speech signal, and the last 14 for the meta-data.

3.3.1. General training of the SCD component

The neural network architecture in Anidjar, Lapidot, Hajaj, Dvir et al. (2021) serves as the base configuration to all the SCD model experiments, where the main change lies in the input layer size, and mainly depends on the text encoding method; e.g., the Facebook pre-trained word-embedding represents each word with 300 features (thus the textual feature set for each SW consists of $300 \cdot 2 = 600$ features), while a Language Agnostic-BERT (Devlin et al., 2018) encoding to each word contains 768 features, thus produces $768 \cdot 2 = 1536$ features. In addition, the model's input layer is fed by 512 additional features of the speech signal, and the 14 meta-data features.

Note that as the dataset is imbalanced, as the proportion between SWs that contain speaker-turns and SWs that do not, is extremely high; we denote by $Split$ the speaker-turns class, i.e. SWs that indeed contain a speaker-turn between the 3rd and 4th words in every SW, and $Same$ for SWs that does not.

Next, we apply a class-weights constraint into the Cross-Entropy (CE) (Qin & Kim, 2019; Zhang & Sabuncu, 2018) loss function with an inverse class weight; that is, the weights are $|Split|^{-1}$ for class $Split$, and $|Same|^{-1}$ for class $Same$. In this manner, the model learns that miss-classification of class $Split$ is harsher than class $Same$. In the Hebrew dataset for instance, the proportion between the $Split$ and $Same$ classes is 1:70.

General training details. Finally, any hidden layer is followed by a ReLU function, and the adaptive Adam (Zhang, 2018) optimizer is used, which computes an adaptive learning rate for the network weights in the training time (starting with an initial value of $\alpha = 10^{-4}$). In addition, a dropout layer is added after any hidden layer of the fully-connected architecture, with a probability value of $p = 0.5$ for 'turning-of' neurons during training time. As aforementioned, the CE loss function is used as the convergence criterion of the neural network during 50 epochs and

applied over a Softmax classification vector of size 2 for the classes $Split$ and $Same$. The rest of the network architecture is similar to the one presented in Table 2 in Anidjar, Lapidot, Hajaj, Dvir et al. (2021).

3.4. Speaker clustering

In SD systems that are unsupervised in their nature, speaker segment-clustering consists of two sub-steps; (i) determination of the number of speakers (i.e. clusters) and (ii) assignment of clustered speaker identities to speaker segments. Note that the first sub-step (number of speakers determination), is out of the scope of this work, and assumed to be a-priori known (Kenai, Asbai, Ouamour, Guerti, & Djeghiour, 2018; Lapidot et al., 2017), and inferred from the conversation in each dataset instead. This use-case can be found in telephone conversations (Lapidot et al., 2017), or in radio programs. As for the assignment of clustered speaker identities to speaker segments, we apply the clustering algorithm from Park and Georgiou (2018), which is based on the Agglomerative Hierarchical Clustering (AHC) (Jaya & Folmer, 2020) algorithm. We used the AHC as it is one of the most acceptable clustering algorithms in the world of SCD and SD. However, more clustering algorithms exist such as the K-Means++ (Vassilvitskii & Arthur, 2006) or subspace clustering (Vidal, 2011). Yet, these algorithms for instance are inappropriate as they suffer from performance degradation whenever the embedding space is a high dimensional one, as in the scenario in this paper (Hämäläinen, Kärkkäinen, & Rossi, 2020; Wang, Liu, So, & Balzano, 2022).

In order to determine which clusters should be merged together (i.e. speaker segments), a measure of dissimilarity between sets of speaker segments is necessary. In this work, we use the Cosine Similarity (from Park and Georgiou (2018)) as a similarity metric for speaker segments as defined in Eq. (3), and the average linkage clustering for the linkage criteria.

$$cosine(x, y) = \frac{x \cdot y}{\|x\| \cdot \|y\|} = \frac{\sum_{i=1}^{n} x_i \cdot y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \cdot \sqrt{\sum_{i=1}^{n} y_i^2}} \tag{3}$$

where x, y from Eq. (3) represent a pair of vectors, i.e. speaker embeddings.

3.5. Speaker Diarization system

After presenting the text-encoding method for the shared-text embedding space, and description of the SCD component and clustering algorithms applied in this subsection, we can attend to the final SD system. The only component being changed throughout the experimental evaluation process is the SCD one; that is, to measure the effectiveness of the suggested SCD component, it is being tested from two main aspects. The first examines it from the speech signal, and different methods for speaker-turn computations such as Wizard-based SCD, fixed-size SCD, and the SCD model presented in Anidjar, Lapidot, Hajaj, Dvir et al. (2021), which is the *S-SCD* (Speech-based SCD) one. The second one examines the SCD models from Anidjar, Lapidot, Hajaj, Dvir et al. (2021), which are the *S-SCD, TS-SCD, TSMD-SCD*, in two ways; (i) signal-based, which is an examination of the contribution of each part in the embeddings vector for SCD, i.e., text, speech, and meta-data, and (ii) multilingual-based, which examine the influence of subsets of languages considered in the training process of the SCD component. In Fig. 2, one can find an illustration of the SD system suggested in this paper. The left shape represents the possible modalities for model training for the purpose of speaker-turns detection, and the right shape demonstrates that a clustering algorithm is applied over the speech segments that are defined by the speaker-turns computation from the previous step (left shape).

Note that the work in Park and Georgiou (2018) employs a one-hot-encoding word-vector to the averaged MFCCs of a speech segment or utterance in order to provide both lexical and acoustic cues to an RNN-based system. However, in this work, we propose to increase the complexity at the feature engineering level. Next, this high-dimensional representation is fed to a rather Fully-Connected neural-network architecture. Still, we did not consider feeding our feature vectors to an RNN topology, since of the following; the three datasets used in this work are ones on which speaker-turns are relatively low. Thus, in order to train an RNN either with LSTM or GRU cells, the maximal sequence might be relatively long, and therefore can lead to three main disadvantages;

(1). As the sequence length grows, it gets harder for an RNN to converge. One reason is that the LSTM cells would have too much input to store in their memory. Thus, we preferred to put our efforts more into the feature engineering process, i.e. better encoding methods, including transformers that outperform RNNs architecture due to their parallelization and attention mechanism, under the assumption that the Fully-Connected architecture would do the rest, as explained in detail in Lippmann (1987) (see Figure 14 in Lippmann (1987)).

(2). As the proportion between the speech and speaker-turns is approximately 1 to 70, i.e. one speaker-turn for every 70 words, it will result in a substantial average training vector for the RNN—as the sentence embedding size is 768, and the speech embedding size is 256, every SW embedding size gets $2 \cdot (768 + 256) = 2048$. With an additional 14 meta-data features, this feature vector size is summed to 2062. Thus, the average modeling of a speech utterance with $70 \cdot 2062$ features is ~145,000 features—which makes the final SCD model to be expensive in terms of computational resources.

(3). From the engineering aspect, an RNN takes much more time to train than a simple DNN one, which leads to an increase in inference time.

4. Datasets

In this work, we target both the SCD and SD problems from the *multilingual* aspect. Thus, three datasets in English, French and Hebrew were collected that differ each from the other, by linguistic characteristics; for instance, a verb in English (West-Germanic language) has at most three possible inflections (*'ed', 'ing', 's'*). In contrast, in Hebrew, the number of inflections for one single verb is numerous since additional such as plural or single form, orders, or tenses—are concatenated to the verb. This fact, for instance, demonstrates the morphological wealth of the Hebrew language, which makes it harder to analyze due to its Semitic structure. Another example of a language difference is gender; In French (Roman language), an adjective depends on whether an object is feminine or masculine. In English, on the other hand, this dependency does not exist. The last two examples are just a drop in the ocean when trying to comprehend the transitions from one language to another. Thus, this work exploits three datasets (Hebrew, English, French) on which the textual aspect differs from each other due to their linguistic structure and characteristics (Semitic, West-Germanic, Roman). These three datasets are discussed separately in Sections 4.1, 4.2, 4.3.

4.1. Hebrew - IFAT group dataset & S2T engine

The Hebrew dataset[4] is composed of 1692 audio-recordings in Hebrew, spreading over 6 months. Each audio-recording duration is approximately 6.5 min, where 92% of the recordings are only containing 2 speakers, and 8% recordings are containing 3–5 speakers. This dataset is heterogeneous and involves 1240 different speakers, and originates from TV shows and radio programs. The proportion of the overlapping speech in the dataset is 5.2%, with 81,301 unique words in total. We used a commercial S2T engine,[5] in order to convert the dataset into automatic transcripts. The Word-Error-Rate (WER) (Wang, Acero, & Chelba, 2003) of the S2T engine is 7.2%.

[4] Visit https://github.com/honeyJarPhD/TextualSpeakerChangeDetection for access to a small subset of the Hebrew conversations. For full dataset access, please e-mail the corresponding author.

[5] Available at: www.almagu.com/voicetotext.

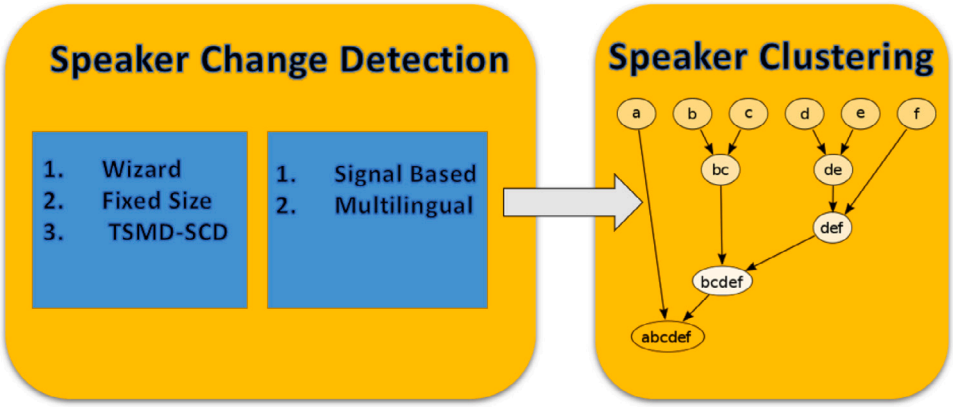

Fig. 2. An illustration of the suggested SD system in this paper. The first component, *and the only one that is changed is the SCD* (the left shape) according to (i) speaker-turn computation methods and (ii) different data types and multilingualism. Then, the Speaker Clustering component is applied to the speech segments computed over the speaker segments extracted from the SCD component. Finally, the SD system is combined with the SCD component and the determination of the number of speakers (clusters) which is a-priori known, in order to construct the dendrogram tree. The dendrogram tree illustration is taken under creative-common license from Hierarchical clustering diagram @ Wikipedia website.

4.2. French - The EPAC project

The EPAC (Esteve, Bazillon, Antoine, Béchet, & Farinas, 2010) corpus[6] is composed of French broadcast conversational speech recordings from French public radio stations (France Inter, France Culture, and Radio France International). It contains about 90 h of speech, related to debates, interviews, and discussions, with manual annotations including segmentation, speaker identity, and transcription. There are 924 different speakers: 264 female (17 h of speech), 656 male (75 h), and four speakers whose gender is not determined. The audio was recorded in studio conditions. The corpus is available on the ELDA catalogue. This dataset contains 40,851 unique words in total.

4.3. English - The ICSI meeting corpus

The ICSI Meeting Corpus (Yella, Stolcke, & Slaney, 2014; Zhang, Wang et al., 2019) is an audio dataset consisting of ~ 70 h of meeting recordings spread over 75 different audio-recordings, that involves 53 different speakers (3–10 participants in each audio). The average length of the audio-recordings is 56 min. The audio-recordings were recorded on close-talking microphones, and are available as SPH or WAV files. In addition, an orthographic transcription, and manual annotation of dialogs and speech quality are available too. The dataset contains 12,263 unique words in total.

5. Evaluation metrics

This section presents the evaluation metrics employed in this work. As we use the textual information in addition to the speech signal, the SCD model is measured differently than the SD system, as follows:

Evaluation Metric for SCD. For each SW in any test set, the SCD model produces a label that determines whether a given test-SW contains a speaker-turn between the 3rd and 4th words. Hence, we measure the SCD model's success using Precision, Recall, and F1-Score computation (Section 5.1).

Diarization Error Rate as an Evaluation Metric. The diarization system's performance is evaluated via a common DER metric, as presented in Section 5.2.

[6] Visit https://catalogue.elra.info/en-us/repository/browse/ELRA-S0305/.

5.1. Evaluation metrics for SCD

Following on from the SW applied over the SCD model, the error computation is done at the word-level. That is, the speaker-turn is considered as such, provided that in a given SW the speaker identity has been changed between the 3rd and 4th words (no matter how long the silent interval is between them). As the error computation is formulated as a binary classification problem, the evaluation metrics are the Precision, Recall and F1-Score, as defined in Eqs. (4), (5) and (6) respectively. The Precision measures the proportion of correct positive identifications, and the Recall measures the proportion of actual positives identified correctly. Finally, the F1-Score is the harmonic mean of Precision and Recall.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{4}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{5}$$

$$F1 - Score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{6}$$

5.2. Diarization error rate as evaluation metric

As described earlier, the AHC algorithm is used for the SD system in Park and Georgiou (2018) and Park et al. (2019). We are using DER as a performance metric for all experiments. To measure the DER metric of a single audio-recording, we employ the md-eval software in RT06S dataset (Fiscus, Ajot, Michel, & Garofolo, 2006) with the forgiveness collar of 0.25 s from each side of the true change point. The DER for a single test-file x is denoted by $der(x)$, and is defined as follows in Eq. (7):

$$der(x) = \frac{False\ Alarm + Miss + Overlap + Confusion}{Length(x)} \tag{7}$$

where False Alarm, Miss, Overlap, Confusion, and Reference Length are defined by:

- **False Alarm:** Length of speech utterances which are supposed to be speech in hypothesis, but are not considered as speech in the reference recording audio-recording.
- **Miss:** Length of speech utterances that are defined as speech in the reference audio-recording, but not in the hypothesis.
- **Overlap:** The length of all the missed speakers in overlapping speech.
- **Confusion:** Length of speech utterances that are assigned to different speakers both in the hypothesis and in the reference audio-recording.

Table 2
Results of the S2T engines when trained over datasets in French and English.

S2T Engines WER results			
Dataset	WER %		
	Substitutions	Deletions	Insertions
ESTER 1 & 2, ETAPE	9.40	7.10	3.70
ICSI Meetings	14.0	20.5	5.80

- **Length(x):** The total length of a test-file x (ground truth) in seconds, **including the overlapping speech**. That is, assuming that a speech segment s involves $z \in \mathbb{N}$ different overlapped speakers, then Length(s) is summed z times to Length(x).

The final reported DER for each experiment in Section 6, and denoted by DER is computed as an averaged der value using Eq. (8):

$$DER = \frac{\sum_{i \in N} Length(x_i) \cdot der(x_i)}{\sum_{i \in N} Length(x_i)} \tag{8}$$

where the size of the test-set in a specific experiment contains the N audio-recordings, $\{x_i\}_{i=1}^{N}$. For a more detailed mathematical formulation, one can find it in Ben-Harush, Ben-Harush, Lapidot, and Guterman (2012).

6. Experimental evaluation

In order to apply the suggested SCD approach in this paper, a S2T engine is required. The ASR systems for English and French languages are based on the Kaldi toolkit (Povey et al., 2011). The acoustic models share the same topology: Hidden Markov Models coupled to Factorized Time-Delay Neural Network (TDNN-f). For both languages, the training data was artificially augmented by speed and volume perturbation.

Actually, the English ASR system is obtained by using the TEDLIUM-3 Kaldi's recipe (tedlium version s5_r3), including the exclusive use of the TEDLIUM-3 audio and textual data (Hernandez, Nguyen, Ghannay, Tomashenko, & Esteve, 2018). TEDLIUM acoustic data are based on TED talk recordings with their manual transcriptions, while the TEDLIUM linguistic data used to train the language models are based on selected subparts of text data allowed during the WMT 2013 machine translation campaign[7]: Europarl v7 corpus, News Commentary, News Crawl. About 152K words compose the vocabulary. A 3-gram language model is first used during the audio decoding to generate word lattices, and a recurrent neural language model is then applied to score them again.

For the French language, the acoustic models were trained on the official data of the ESTER1 (Galliano et al., 2006), ESTER2 (Galliano et al., 2005), ETAPE French evaluation campaigns (Gravier et al., 2012). The language models were trained on the manual transcription of this audio data and newspaper articles and news web crawling. The vocabulary contains 160K words. The EPAC dataset was not used to train this model. The WER of the TEDLIUM-based English S2T engine for the ICSI meeting dataset is 40.3%, while the broadcast news-based French S2T engine reaches a WER of 20.2% on the EPAC conversational broadcast dataset. In Table 2 we detail the nature of ASR errors of the S2T engines.

Next, the sequel of this section is dedicated to validating our hypothesis that the multilingual framework suggested in this paper results in an accurate and efficient speaker-turn estimation component to construct a powerful SD system. We divide our experimental evaluation into the following three parts;

S2T Based SCD. The objective of the first experimental set is to examine different approaches for speaker-turn computation, in order to comprehend its influence over SD system. The goal is to test the influence of these approaches over the SD system presented by Park and Georgiou (2018).

Applying SCD Using Tri-lingual Dataset. The next experimental evaluation set tries to grasp the relationship between different components, i.e. text, speech, meta-data, and their combinations over the SCD system as in Anidjar, Lapidot, Hajaj, Dvir et al. (2021). Then, the multilinguality of the SCD component and how it influences the SD system is examined over three languages: Hebrew, French, and English. Finally, we show how subsets of languages are influencing the performance of the SD system in Park and Georgiou (2018). It is important to clarify that the textual information and meta-data features are applied only for the SCD models, while the clustering algorithm to compute the speaker labels for SD, are applied over the speech embeddings only.

6.1. SCD component for Speaker Diarization

Previous works (India Massana et al., 2017; Park & Georgiou, 2018; Park et al., 2019) have shown how to exploit the speech signal in order to construct an SD system, followed by an SCD component. However, only a few exploited the text signal, either by a wizard annotation dataset or a S2T engine. Thus, this section is dedicated to exploring different SCD components from the speech signal only, in order to appreciate how the text signal improves the speaker-turn computation (Section 6.2). Thus, the following approaches for the speaker-turn computation are examined; (i) a wizard-based computation, which is a perfect SCD; (ii) a short-segment extraction of 1.5 s, which is derived from the SD model presented in Silnova et al. (2020); (iii) a BIC-based model that detects speaker-turns; (iv) a wizard-based computation that requires training of an SCD system having a perfect S2T engine, specifically the one presented by Anidjar, Lapidot, Hajaj, Dvir et al. (2021); (v) training of the SCD system over a S2T based dataset; Finally, we present both the SCD and SD results for each of these experiments.

We point out that this part of the experimental evaluation is being processed and analyzed over the ICSI Meeting Corpus (Section 4.3) only. While the last two experiments in this section require a pre-training of an SCD component, the first two experiments in this section do not require any. Thus, for all of the next four experiments, the ICSI dataset is *randomly* split into training and test sets, such that 60 out of the 75 meetings (80% in total) are serving for training, and the results of all these four experiments are reported over the remaining test-set of 15 meetings, (20% in total).

1. ***Wizard-Based Speaker Diarization*** - A baseline for computation of speaker-turns, which is the ideal situation, is based on the wizard version of the ICSI dataset. Thus, the speech segments are computed perfectly as the wizard annotation dataset is provided with this information. As for speech overlapping, we consider any overlap starting and ending as a speaker-turn.
   SCD. Each of the 15 test audio-recordings of the ICSI dataset was split into a consecutive set of speaker-turns using the wizard annotation dataset information. Any starting or ending is considered as speaker overlap, but as long as the speaker identity has not changed, no speaker-turn is considered, even if the non-speech part of the speaker segment is very long. Under these assumptions, every two consecutive speaker segments are tagged as $Split$ if there is a speaker-turn, or $Same$ otherwise. We denote this model by *WIZARD*.
   SD. The clustering algorithm from Park and Georgiou (2018) is applied over the speaker segments of different lengths in the 15 test-set audio-recordings, encoded into a 256-dimensional embedding, the same as in the previous case.
2. ***Speaker Diarization with Short Segments*** - As a simple baseline for computation of speaker-turns, we used the same idea as presented in Silnova et al. (2020), which explored probabilistic embeddings for speaker diarization, using Probabilistic Linear

[7] Visit https://www.statmt.org/wmt13/translation-task.html.

Discriminant Analysis (PLDA) (Silnova et al., 2020), speech embeddings are extracted from very short (1.5 s) speech segments, followed by clustering of the embeddings with respect to speaker identity.

SCD. Each of the 15 test audio-recordings of the ICSI dataset was split into a consecutive set of 1.5 s that sums up the total length of each audio recording. We denote this model by *SCD-1.5*.

SD. The clustering algorithm from Park and Georgiou (2018) is applied over the 1.5 seconds-long segments in the 15 test-set audio-recordings, where each such segment is encoded into a 256-dimensional embedding, as explained in Section 3.2.

3. ***BIC-Based Speaker Diarization*** - In Chen, Gopalakrishnan, et al. (1998), by modeling audio recordings as a Gaussian process in the cepstral space, is suggested a maximum likelihood approach for acoustic speaker-turn estimation, based on the BIC (Chen et al., 1998, 2020) criterion. This approach examines the speech signal only. Therefore, and in order to justify the exploitation of textual information, we used the system suggested by Chen et al. (1998) for this purpose.

   SCD. Each of the 15 test audio-recordings of the ICSI dataset was split into a consecutive set of speaker-turns using the BIC criterion as suggested in Chen et al. (1998). We denote this model by *SCD-BIC*.

   SD. The clustering algorithm from Park and Georgiou (2018) is applied over the speaker segments of different lengths in the 15 test-set audio-recordings, encoded into a 256-dimensional embedding, the same as in the previous case.

4. ***Wizard-Based SCD for Speaker Diarization*** - The wizard-based SCD scenario is one on which the SCD component is trained over the 60 training audio-recordings of the wizard annotation dataset, as earlier explained in Section 6.1, i.e., a perfect S2T engine was applied, but the true change detection was not used. The goal is to measure how effective the SCD components are 'in-the-wild', where the ground truth is already known. It should serve as an optimistic upper bound of the SD compared with the text-based SCD module.

   SCD. The 60 training-set audio-recordings of the ICSI dataset were fed into the neural network architecture suggested by Anidjar, Lapidot, Hajaj, Dvir et al. (2021), specifically in the *S-SCD* model from the $Power - Set$ experiment, which is an SCD component that is based on the speech signal only (Section VII-A in Anidjar, Lapidot, Hajaj, Dvir et al. (2021)). We denote this model by *WIZARD-SCD*.

   SD. The 15 test-set audio-recordings were evaluated through the *S-SCD* model in order to get the speaker-turns in the wizard annotation dataset (test-set only). Next, the clustering algorithm from Park and Georgiou (2018) is applied over the speaker segments of different lengths according to the SCD output, in the 15 test-set audio-recordings.

5. ***S2T-Based Speaker Diarization*** - The S2T-based SCD scenario is a trifle distinct. In this case, the S2T transcript provides each word's starting and ending time, which means that the speech embeddings can be deduced and computed using this word-level information. An SCD component is trained over the training set, with respect to the *S-SCD* model.

   SCD. The 60 training-set audio-recordings of the ICSI dataset were fed into the *S-SCD* model, where the speech embeddings are computed at the word-level. We denote this model by *S2T-SCD*, since it is trained over a S2T-based dataset (rather than a wizard one).

   SD. The 15 test-set audio-recordings were evaluated through the *S-SCD* model in order to get the speaker-turns in the S2T-based dataset (test-set only). Next, the clustering algorithm from Park and Georgiou (2018) is applied over the word-level speech embeddings in the 15 test-set audio-recordings.

Table 3
Results table of the experimental evaluation set in Section 6.1 for the speech-signal based SCD component, and its influence over a diarization system whenever the dataset conditions are changed.

Speech-based SCD model list					
Entry	Model name	**SCD**			**SD**
		Precision	Recall	F1-Score	DER
(1)	**WIZARD**	100.0	100.0	100.0	10.12
(2)	**SCD-1.5**	30.48	67.23	41.94	38.07
(3)	**SCD-BIC**	82.14	83.65	82.88	19.53
(4)	**WIZARD-SCD**	89.25	88.93	89.09	15.92
(5)	**S2T-SCD**	87.45	86.22	86.83	17.84

The results of the different SCD approaches that are described in this section are presented and summarized in Table 3. As expected, the WIZARD model achieves ideal and perfect results as an SCD component, which complies well with its DER result whenever the speaker-turns are computed by using the wizard annotation dataset. On the other hand, whenever manual alignment is getting into the picture, and the wizard annotation dataset is trained and tested over the SCD component (WIZARD-SCD), the SCD results are getting lower than the previous model, as well as the DER. Yet, the results in Ent. (4) remain relatively competitive to an *ideal* and *perfect* world-state as in Ent.(1), thus one can conclude that the SCD component works fine when trained over the speech signal only.

In Ent. (5), one can see that the S2T-SCD model is not left significantly behind the WIZARD-SCD (Ent. (4)) one. We hypothesize that this degradation is caused due to the WER of the S2T engine that aligned the ICSI dataset into a continuous text, which affects both the SCD and SD performance. Finally, Ent.(2) represents the most simple baseline without an SCD component at all, but rather a baseline heuristic to determine the speaker-turns. As overlapping speech is not considered at the SCD stage in the SCD-1.5 model (as it has no SCD component), we can expect to this relatively low results in the SCD part, which is followed by a higher DER value, as it gets hard to compensate these small speech segments. Moreover, the embeddings module is trained for much larger speech utterances. Therefore short-segments are not encoded well enough as the voice-wave and non-speech parts repetition is much higher than the rest of the approaches.

As for the SCD-BIC model (Ent. (3)), one can note that using the speech signal solely yields competitive results, when compared to a S2T based models (as in Ent. (5)). Nevertheless, Section 6.2 will demonstrate how to improve these results, by exploiting the textual information and the meta-data.

6.2. Applying SCD over SD using Tri-lingual dataset

In order to acknowledge the strength of the text and meta-data signals over an SCD system, as well as the efficiency of such a model over multiple datasets, this section is dedicated to the multilinguality and subsets of signals examination of the SCD model over the SD system. Specifically, we extend the previous experimental evaluation set into three languages instead of only one. Next, we show how powerful the text signal is for an SCD system, and how the addition of meta-data improves the SCD system. Finally, we present the SCD and SD results for each of the experiments.

We point out that this part of the experimental evaluation is being processed and analyzed over all three datasets in Hebrew, French and English, as this section is dedicated mainly to the text signal and how it is affected by the multilinguality. Eventually, the three experiments in this section extend the experiments from Section 6.1, where the main difference is that the Tri-lingual dataset is being examined, as well as its subsets. Thus, for all of the next three experiments, the Tri-lingual dataset is *randomly* split into training and test sets, such that 80% are serving for training-set, and the results of all these three experiments are reported over the remaining 20% for the test-set, as follows;

Hebrew. The IFAT Group dataset consists of 1692 audio-recordings and their S2T transcripts. Thus, 1354 of them served for training-set, and the remaining 338 for test-set.

French. The EPAC dataset consists of 129 audio-recordings. Thus, 104 of them served for training-set, and the remaining 25 for test-set.

English. The ICSI is being split the same as in Section 6.1, i.e. 60 out of 75 audio-recordings served for training-set, and the remaining 15 for test-set.

1. Speech-Based Speaker Diarization—The goal of this section is nearly the same as the one in Section 6.1, with a slight modification. This time, the SCD component is trained separately over each of the Hebrew and French languages, using the S2T transcripts of each dataset that provide the starting and ending times of each word. Clearly, an SCD component is trained over the training-set of each language, using the *S-SCD* model. **SCD.** Two SCD components are trained over the training sets in Hebrew and French, and fed into the *S-SCD* model, where the embeddings are computed at the word-level. The result is two SCD components for each language, with an additional SCD component already trained for English. We denote these models by *SSCD-HEB* and *SSCD-FR* respectively. The third model for the English language denotes by *SSCD-EN* in this section for consistency purposes. **SD.** The test-set of each language was evaluated using its corresponding *S-SCD* model that was trained over the dataset in the same language, in order to get the speaker-turns in the S2T-based dataset (test-set only). Next, the clustering algorithm from Park and Georgiou (2018) is applied over the word-level embeddings in the test-sets of the Hebrew and French languages.
2. Text-Speech-Based Speaker Diarization—This section extends the previous one, as the text signal is considered along with the speech one. Thus, we use the neural network architecture suggested by Anidjar, Lapidot, Hajaj, Dvir et al. (2021), specifically in the *TS-SCD* model from the *Power – Set* experiment, which is an SCD component that is based on the speech and text signals only (Section VII-A in Anidjar, Lapidot, Hajaj, Dvir et al. (2021)). **SCD.** Three SCD components are trained over the training sets in Hebrew, French and English, and fed into the *TS-SCD* model, where the embeddings are computed at the word-level. The result is three SCD components for each language. We denote these models by *TSSCD-HEB*, *TSSCD-FR* and *TSSCD-EN* for Hebrew, French and English respectively. **SD.** The test-set of each language was evaluated using its corresponding *TS-SCD* model that was trained over the dataset in the same language, in order to get the speaker-turns in the S2T-based dataset (test-set only). Next, the clustering algorithm from Park and Georgiou (2018) is applied over the word-level embeddings in the test-sets of the three languages.
3. TSMD-Based Speaker Diarization—Following the *TSMD-SCD* model in Anidjar, Lapidot, Hajaj, Dvir et al. (2021), this section presents the SCD component as a combination of the speech, text, and meta-data signals from the multilingual aspect. That is, each dataset is examined both jointly and separately as part of the possible 7 subsets, followed by their model names. For each of the following 7 models, the TSMD-SCD model and the SD system are trained and tested on;

    - **TSMD-HEB.** The Hebrew dataset.
    - **TSMD-FR.** The French dataset.
    - **TSMD-EN.** The English dataset.
    - **TSMD-HF.** The Hebrew and French datasets.
    - **TSMD-HN.** The Hebrew and English datasets.
    - **TSMD-FN.** The French and English datasets.
    - **TSMD-HFN.** The Hebrew, French, and English datasets.

    SCD. Seven SCD models are trained over the training sets in Hebrew, French and English, depending on the training model from the list above. Then, the training set is fed into the *TSMD-SCD* model, where the embeddings are computed at the word-level. The result is seven SCD components for each language. **SD.** The test-set of each subset of languages was evaluated using its corresponding *TSMD-SCD* model that was trained over the dataset in the same language, in order to get the speaker-turns in the S2T-based dataset (test-set only). Next, the clustering algorithm from Park and Georgiou (2018) is applied over the word-level embeddings in the test-sets of the seven subsets of languages.

The results of the experimental evaluation set in Section 6.2 are presented and summarized in Table 4. We discuss the results at the following different levels;

S-SCD Models. Entries (1)–(3) represent the speech-based model for each language separately. As the embedding extractor is trained on English, we can expect to degradation of performance when the S-SCD model is trained on Hebrew (Ent. (1)) or French (Ent. (2)).

TS-SCD Models. Entries (4)–(6) represent the hybrid model (TS-SCD), which is based on the speech and text signals for each language separately. As the textual information is encoded through the shared word-embedding linear space of the Language Agnostic-BERT model, the similarity between correlated words gets higher, despite the remoteness between languages such as Hebrew (Semitic), English (West-Germanic) and French (Roman). Thus, we can expect to better SCD results, which affect the similarity of the DER results of the three models.

TSMD Models. Entries (7)–(13) represent the full hybrid model (TSMD-SCD Anidjar, Lapidot, Hajaj, Dvir et al., 2021), which is based on text, speech, and meta-data features, trained on every subset of languages among Hebrew, English and French. As expected, when the TSMD model is trained on every language separately (Entries (7)–(9)), one can note the outperformance of this hybrid over the previous two versions (S-SCD and TS-SCD), both for the SCD and SD components, while still the textual part is extracted using Language Agnostic-BERT. Next, whenever the TSMD model is trained over two languages (Entries (10)–(12)), one can note that the performance of the SCD component is lower, compared to single-language-based models (Entries (7)–(9)), and correspondingly the DER's gets a bit higher. However, whenever the TSMD model is trained over Hebrew, English and French (Entry. (13)), the results are competitive compared to dual-language based models (Entries (10)–(12)) and justify the representation of three different languages in one single model.

Overall. Taking into account the feature-engineering process, one can conclude that the textual information improves the performance of an SCD component, and as such, of an SD system. As suggested in the TSMD-SCD approach, aggregation of meta-data features is showing preponderance, mainly in the Recall and DER indicators. In addition, in order to conclude from Tables 4 and 5 that the multilingual version of our framework is competitive enough whenever a model is trained for one single language only, we provide a short inference from these two tables for each language separately, as follows: **Hebrew.** The Hebrew language is represented as a monolingual-based model in Table 4 in Ent.(7), with a 10.33 DER value. On the other hand, in Table 5, one can note that the DER value of a model that was trained on a multilingual dataset including Hebrew, is competitive to the monolingual DER value achieved for Hebrew; (i) 10.36 DER for Hebrew and French (Ent.(1)); (ii) 10.29 DER for Hebrew and English (Ent. (2)); and (iii) 10.32 DER for Hebrew, French and English (Ent. (4)). **French.** The French language is represented as a monolingual-based model in Table 4 in Ent.(8), with a 12.11 DER value. On the other hand, in Table 5, one can note that the DER value of a model that was trained on a multilingual dataset including French, is competitive to the monolingual DER value achieved for French; (i) 12.14 DER for French and Hebrew (Ent. (1));

Table 4
Results table of the experimental evaluation set in Section 6.2, for the SCD component and its variants with respect to the speech, text and meta-data signals, and how these variants over subsets of the Tri-lingual dataset are influencing over a diarization system.

Multilingual model list					
Entry	Model name	**SCD**			**SD**
		Precision	Recall	F1-Score	DER
(1)	**SSCD-HEB**	85.20	83.60	84.39	17.57
(2)	**SSCD-FR**	88.32	86.47	87.38	16.53
(3)	**SSCD-EN**	87.45	86.22	86.83	17.84
(4)	**TSSCD-HEB**	97.46	91.37	94.32	11.73
(5)	**TSSCD-FR**	97.12	94.26	95.66	12.67
(6)	**TSSCD-EN**	97.49	96.98	97.23	11.45
(7)	**TSMD-HEB**	97.52	96.41	96.96	10.33
(8)	**TSMD-FR**	97.78	96.23	96.99	12.11
(9)	**TSMD-EN**	97.65	97.67	97.66	10.28
(10)	**TSMD-HF**	94.18	93.01	93.59	11.42
(11)	**TSMD-HN**	95.01	93.42	94.25	11.75
(12)	**TSMD-FN**	95.32	93.74	94.52	11.83
(13)	**TSMD-HFN**	93.81	92.90	93.35	11.64

Table 5
DER results for the multilingual models in Table 4, separated by each of the languages that are used to train each of the multilingual models. For ease of reading, Entry (5) represents the DER results column of the monolingual TSMD models from Entries (7)–(9) in Table 4.

Separated-Language DER results				
Entry	Model name	DER		
		Hebrew	French	English
(1)	**TSMD-HF**	10.36	12.14	–
(2)	**TSMD-HN**	10.29	–	10.38
(3)	**TSMD-FN**	–	12.23	10.27
(4)	**TSMD-HFN**	10.32	12.24	10.36
(5)	**Monolingual**	10.33	12.11	10.28

(ii) 12.23 DER for French and English (Ent. (3)); and (iii) 12.24 DER for French, Hebrew and English (Ent. (4)). **English.** The English language is represented as a monolingual-based model in Table 4 in Ent.(9), with a 10.28 DER value. On the other hand, in Table 5, one can note that the DER value of a model that was trained on a multilingual dataset including English, is competitive to the monolingual DER value achieved for English; (i) 10.38 DER for English and Hebrew (Ent. (2)); (ii) 10.27 DER for English and French (Ent. (3)); and (iii) 10.36 DER for English, Hebrew and French (Ent. (4)).

Next, we present in Table 5 the DER results for the multilingual models (Entries (10)–(13) in Table 4), separated by each of the languages. One can note that the difference between the DER results of the monolingual models (Entry (5) in Table 5) and the DER results when separated for each language (Entries (1)–(4) in Table 5) is statistically negligible.

ICSI Dataset. In Table 4 we have shown the performance improvement whenever modalities are added to the SCD model, and the manner they are influencing over an SD system. Taking the ICSI dataset as a test case, one can note the performance upgrade from these modalities. Namely, while Ent. (3) represents a SD system that is both trained and tested over the ICSI dataset and achieves 17.84 DER, we can note that adding textual information in Ent. (6) results in 11.45 DER. Next, when meta-data is supplemented as well, the results of the triple-modal-based SD system achieve 10.28 DER (Ent. (9)). As the ICSI is a well-known (Yella et al., 2014; Zhang, Wang et al., 2019) dataset that served for several previous SD works, we report in Table 6 the DER results over the ICSI dataset, when comparing our approach and its evolutional models, and the methods suggested from the following two previous works;

Table 6
DER results for the ICSI dataset.

DER results on the ICSI dataset		
Entry	Model name	DER
(1)	**SSCD-EN**	17.84
(2)	**TSSCD-EN**	11.45
(3)	**TSMD-EN**	**10.28**
(4)	**MFCC** (Yella et al., 2014)	18.40
(5)	**ANN + MFCC** (Yella et al., 2014)	15.10
(6)	**Att-v2s - VB** (Lin et al., 2020)	18.98
(7)	**Att-v2s + VB** (Lin et al., 2020)	18.44

- **Artificial Neural Network (ANN) Features for SD.** In Yella et al. (2014), multi-hidden-layer ANN was trained in order to determine whether two given speech utterances belong to the same speaker or different ones, by using a shared transform of input features that are fed into a bottleneck layer. Next, the bottleneck layer activations are used as features, either by themselves or combined with MFCC features.
- **Self-Attentive Similarity Measurement Strategies in SD.** In Lin, Hou, and Li (2020), the authors presented a method based on the mechanism of self-attention (Devlin et al., 2018), which instead of focusing on local correlation, it searches for the similarity between speaker embeddings.

It can be seen from Table 6 that not only is the method presented in this paper competitive with the other two methods presented in Lin et al. (2020) and Yella et al. (2014), but also significantly improves their performance on the ICSI dataset when textual information and metadata modalities are added to the model.

7. Conclusions and future work

In this paper, we have shown the benefits of exploiting textual information and speech analysis in order to construct a multilingual hybrid model for the SCD problem, and how to apply it over an SD system. With datasets in Hebrew, French, and English, we have shown how to use the Language Agnostic-BERT model, in order to construct a CLV mechanism that (i) represents these three languages at the same time; and (ii) serves to train a single SCD model, which is competitive enough instead of training two or three distinct models for each language separately.

We compared several SCD approaches, including a frequently used BIC-based approach for SCD. The proposed system gives more than 5% absolute improvement in SCD precision, which led to more than 1.5% absolute improvement in DER. In addition, we showed that our architecture is robust to a high WER; the WER on the ICSI Meetings English database the WER was about 40%, the degradation in DER with respect to French and Hebrew databases was about 1% only (see Table 4). In addition, it was shown that textual information led to former improvement in SCD and SD. While adding a meta-data gives an additional gain. Each modality (speech, text, and meta-data) has complementary information that results in a total improvement of the SD system (8% – 10% absolute improvement in precision; 5% – 7% absolute improvement in DER).

One may consider that our approach has a drawback, as in a closed set of languages, there is a gap when a removal or supplement of one (or more) language to the model is needed (assuming that there exists an S2T engine that recognizes it). Nevertheless, our framework can be easily adapted in order to address this gap, since the Agnostic-BERT model supports over 100 different languages that cover a respectable percentage of the spoken languages around the world. Moreover, the multilingual SCD that is trained over the three languages presented in this paper shows only a slight degradation in terms of SCD precision and no degradation in terms of DER. Especially, even for the Hebrew

language, which is essentially different from the other two, there was no degradation in terms of DER. Additional future work might be the development of an end-to-end system that detects the spoken language in an audio-recording, apart from detecting who spoke and when.

CRediT authorship contribution statement

Or Haim Anidjar: Conceptualization, Methodology, Data curation, Visualization, Resources, Investigation, Software, Writing – original draft. **Yannick Estève:** Formal analysis, Writing – review. **Chen Hajaj:** Formal analysis, Writing – review, Supervision, Methodology. **Amit Dvir:** Formal analysis, Writing – review, Supervision, Methodology. **Itshak Lapidot:** Validation, Investigation, Writing – review & editing, Data curation, Supervision, Investigation, Methodology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The authors wish to thank IFAT Group which provided the Hebrew dataset. In addition, this work was supported by the Ariel Cyber Innovation Center in conjunction with the Israel National Cyber directorate in the Prime Minister's Office, and Ariel Data Science and Artificial Intelligence Research Center.

References

Anidjar, O. H., Lapidot, I., Hajaj, C., & Dvir, A. (2021). A thousand words are worth more than one recording: Word-embedding based speaker change detection. In *Proc. interspeech 2021* (pp. 3121–3125).

Anidjar, O. H., Lapidot, I., Hajaj, C., Dvir, A., & Gilad, I. (2021). Hybrid speech and text analysis methods for speaker change detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *29*, 2324–2338.

Artetxe, M., Labaka, G., & Agirre, E. (2016). Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *CEMNLP* (pp. 2289–2294).

Barry, J., Boschee, E., Freedman, M., & Miller, S. (2020). SEARCHER: Shared embedding architecture for effective retrieval. In *CLSSTS2020* (pp. 22–25).

Ben-Harush, O., Ben-Harush, O., Lapidot, I., & Guterman, H. (2012). Initialization of iterative-based speaker diarization systems for telephone conversations. *IEEE Transactions on Audio, Speech, and Language Processing*, *20*(2), 414–425.

Bhowmik, B., Tripura, T., Hazra, B., & Pakrashi, V. (2020). Robust linear and non-linear structural damage detection using recursive canonical correlation analysis. *Mechanical Systems and Signal Processing*, *136*, Article 106499.

Brychcín, T. (2020). Linear transformations for cross-lingual semantic textual similarity. *Knowledge-Based Systems*, *187*, Article 104819.

Camacho-Collados, J., Doval, Y., Martínez-Cámara, E., Espinosa-Anke, L., Barbieri, F., & Schockaert, S. (2020). Learning cross-lingual word embeddings from Twitter via distant supervision. In *AAAI conference on web and social media, Vol. 14* (pp. 72–82).

Carletta, J., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., et al. (2005). The AMI meeting corpus: A pre-announcement. In *International workshop on machine learning for multimodal interaction* (pp. 28–39). Springer.

Chen, S., Gopalakrishnan, P., et al. (1998). Speaker, environment and channel change detection and clustering via the bayesian information criterion. In *Proc. DARPA broadcast news transcription and understanding workshop, Vol. 8* (pp. 127–132). Virginia, USA.

Chen, L., Lee, K. A., He, L., & Soong, F. K. (2020). ON early-stop clustering for speaker diarization. In *Proc. odyssey 2020 the speaker and language recognition workshop* (pp. 110–116).

Conneau, A., & Lample, G. (2019). Cross-lingual language model pretraining. In *Advances in neural information processing systems* (pp. 7059–7069).

Deléglise, P., Esteve, Y., Meignier, S., & Merlin, T. (2009). Improvements to the LIUM french ASR system based on CMU sphinx: what helps to significantly reduce the word error rate? In *Tenth annual conference of the international speech communication association*.

Deschamps-Berger, T. (2021). Emotion recognition in emergency call centers: The challenge of real-life emotions. In *2021 9th international conference on affective computing and intelligent interaction workshops and demos (ACIIW)* (pp. 1–5). IEEE.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL* (pp. 6321–6325).

El-Kassas, W. S., Salama, C. R., Rafea, A. A., & Mohamed, H. K. (2021). Automatic text summarization: A comprehensive survey. *Expert Systems with Applications*, *165*, Article 113679.

Esteve, Y., Bazillon, T., Antoine, J.-Y., Béchet, F., & Farinas, J. (2010). The EPAC corpus: Manual and automatic annotations of conversational speech in french broadcast news.. In *LREC*. Citeseer.

Fang, S.-H., Tsao, Y., Hsiao, M.-J., Chen, J.-Y., Lai, Y.-H., Lin, F.-C., et al. (2019). Detection of pathological voice using cepstrum vectors: A deep learning approach. *Journal of Voice*, *33*(5), 634–641.

Feng, F., Yang, Y., Cer, D., Arivazhagan, N., & Wang, W. (2020). Language-agnostic bert sentence embedding. arXiv preprint arXiv:2007.01852.

Fini, E., & Brutti, A. (2020). Supervised online diarization with sample mean loss for multi-domain data. In *ICASSP* (pp. 7134–7138).

Firat, O., Cho, K., & Bengio, Y. (2016). Multi-way, multilingual neural machine translation with a shared attention mechanism. In *North American chapter of the association for computational linguistics: human language technologies* (pp. 866–875).

Firc, A., & Malinka, K. (2022). The dawn of a text-dependent society: deepfakes as a threat to speech verification systems. In *Proceedings of the 37th ACM/SIGAPP symposium on applied computing* (pp. 1646–1655).

Fiscus, J. G., Ajot, J., Michel, M., & Garofolo, J. S. (2006). The rich transcription 2006 spring meeting recognition evaluation. In *International workshop on machine learning for multimodal interaction* (pp. 309–322). Springer.

Fobbe, E. (2022). Forensic linguistics. *Zeitschrift Für Polizeiwissenschaft Und Polizeiliche Praxis*, *19*(5), 31–39.

Fu, Z., Xian, Y., Geng, S., Ge, Y., Wang, Y., Dong, X., et al. (2020). ABSent: Cross-lingual sentence representation mapping with bidirectional GANs. In *AAAI*.

Fujita, Y., Kanda, N., Horiguchi, S., Xue, Y., Nagamatsu, K., & Watanabe, S. (2019). End-to-end neural speaker diarization with self-attention. In *2019 IEEE automatic speech recognition and understanding workshop (ASRU)* (pp. 296–303). IEEE.

Fujita, Y., Watanabe, S., Horiguchi, S., Xue, Y., & Nagamatsu, K. (2020). End-to-end neural diarization: Reformulating speaker diarization as simple multi-label classification. arXiv preprint arXiv:2003.02966.

Galibert, O. (2013). Methodologies for the evaluation of speaker diarization and automatic speech recognition in the presence of overlapping speech. In *INTERSPEECH* (pp. 1131–1134).

Galliano, S., Geoffrois, E., Gravier, G., Bonastre, J.-F., Mostefa, D., & Choukri, K. (2006). Corpus description of the ester evaluation campaign for the rich transcription of french broadcast news. In *In proceedings of the 5th international conference on language resources and evaluation (LREC 2006)*.

Galliano, S., Geoffrois, E., Mostefa, D., Choukri, K., Bonastre, J.-F., & Gravier, G. (2005). The ESTER phase II evaluation campaign for the rich transcription of french broadcast news. In *Ninth European conference on speech communication and technology*.

Gehring, J., Auli, M., Grangier, D., Yarats, D., & Dauphin, Y. N. (2017). Convolutional sequence to sequence learning. In *ICML* (pp. 1243–1252).

Gravier, G., Adda, G., Paulson, N., Carré, M., Giraudel, A., & Galibert, O. (2012). The ETAPE corpus for the evaluation of speech-based TV content processing in the french language. In *LREC-eighth international conference on language resources and evaluation* (p. na).

Hämäläinen, J., Kärkkäinen, T., & Rossi, T. (2020). Improving scalable K-means++. *Algorithms*, *14*(1), 6.

Hernandez, F., Nguyen, V., Ghannay, S., Tomashenko, N., & Esteve, Y. (2018). TED-LIUM 3: twice as much data and corpus repartition for experiments on speaker adaptation. In *International conference on speech and computer* (pp. 198–208). Springer.

India Massana, M. À., Rodríguez Fonollosa, J. A., & Hernando Pericás, F. J. (2017). LSTM neural network-based speaker segmentation using acoustic and language modelling. In *INTERSPEECH* (pp. 2834–2838).

Jaya, I. G. N. M., & Folmer, H. (2020). Identifying spatiotemporal clusters by means of agglomerative hierarchical clustering and Bayesian regression analysis with spatiotemporally varying coefficients: methodology and application to dengue disease in Bandung, Indonesia. *Geographical Analysis*.

Jin, X.-B., Yang, N.-X., Wang, X.-Y., Bai, Y.-T., Su, T.-L., & Kong, J.-L. (2020). Hybrid deep learning predictor for smart agriculture sensing based on empirical mode decomposition and gated recurrent unit group model. *Sensors*, *20*(5), 1334.

Juan, S. S., Ismail, M. F. C., Ujir, H., & Hipiny, I. (2020). Language modelling for a low-resource language in sarawak, Malaysia. In *Advances in electronics engineering* (pp. 147–158). Springer.

Kang, W., Roy, B. C., & Chow, W. (2020). Multimodal speaker diarization of real-world meetings using D-vectors with spatial features. In *ICASSP* (pp. 6509–6513). IEEE.

Katharopoulos, A., Vyas, A., Pappas, N., & Fleuret, F. (2020). Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning* (pp. 5156–5165). PMLR.

Kenai, O., Asbai, N., Ouamour, S., Guerti, M., & Djeghiour, S. (2018). Speaker diarization and detection system using a priori speaker information. In *2018 2nd international conference on natural language and speech processing (ICNLSP)* (pp. 1–6). IEEE.

Kunešová, M., Hrúz, M., Zajíc, Z., & Radová, V. (2019). Detection of overlapping speech for the purposes of speaker diarization. In *International conference on speech and computer* (pp. 247–257). Springer.

Landini, F., Profant, J., Diez, M., & Burget, L. (2022). Bayesian HMM clustering of x-vector sequences (VBx) in speaker diarization: theory, implementation and analysis on standard tasks. *Computer Speech and Language*, *71*, Article 101254.

Lapidot, I., Shoa, A., Furmanov, T., Aminov, L., Moyal, A., & Bonastre, J.-F. (2017). Generalized viterbi-based models for time-series segmentation and clustering applied to speaker diarization. *Computer Speech and Language*, *45*, 1–20.

Le, N., & Odobez, J.-M. (2019). Improving speech embedding using crossmodal transfer learning with audio-visual data. *Multimedia Tools and Applications*, *78*(11), 15681–15704.

Li, W., Liu, K., Zhang, L., & Cheng, F. (2020). Object detection based on an adaptive attention mechanism. *Scientific Reports*, *10*(1), 1–13.

Lin, Q., Hou, Y., & Li, M. (2020). Self-attentive similarity measurement strategies in speaker diarization.. In *INTERSPEECH* (pp. 284–288).

Lippmann, R. (1987). An introduction to computing with neural nets. *IEEE Assp Magazine*, *4*(2), 4–22.

Litvinov, D. M. (2021). Speech analytics architecture for banking contact centers. In *10th annual international scientific and practical conference named after AI Kitov information technologies and mathematical methods in economics and management, IT and MM-CEUR workshop proceedings, Vol. 2830* (pp. 227–239).

Liu, X., Duh, K., Liu, L., & Gao, J. (2020). Very deep transformers for neural machine translation. arXiv preprint arXiv:2008.07772.

Liu, D., & Kubala, F. (1999). Fast speaker change detection for broadcast news transcription and indexing. In *Sixth European conference on speech communication and technology*.

Lu, L., & Zhang, H.-J. (2002). Speaker change detection and tracking in real-time news broadcasting analysis. In *Proceedings of the tenth ACM international conference on multimedia* (pp. 602–610).

von Lühmann, A., Li, X., Müller, K.-R., Boas, D. A., & Yücel, M. A. (2020). Improved physiological noise regression in fNIRS: A multimodal extension of the general linear model using temporally embedded canonical correlation analysis. *NeuroImage*, *208*, Article 116472.

Luo, C., Wu, X., Zheng, T. F., & Wang, L. (2010). Segmentation-based method for text-dependent speaker recognition in embedded applications. *APSIPA ASC*.

Ma, C., Li, W., Cao, J., Du, J., Li, Q., & Gravina, R. (2020). Adaptive sliding window based activity recognition for assisted livings. *Information Fusion*, *53*, 55–65.

Meng, Z., Mou, L., & Jin, Z. (2017). Hierarchical RNN with static sentence-level attention for text-based speaker change detection. In *Conference on information and knowledge management* (pp. 2203–2206).

Nozza, D., Bianchi, F., & Hovy, D. (2020). What the [mask]? making sense of language-specific BERT models. arXiv preprint arXiv:2003.02912.

Ortega, J. E., Mamani, R. C., & Cho, K. (2020). Neural machine translation with a polysynthetic low resource language. *Machine Translation*, *34*(4), 325–346.

Pan, Y., Yao, T., Li, Y., & Mei, T. (2020). X-linear attention networks for image captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10971–10980).

Park, T. J., & Georgiou, P. G. (2018). Multimodal speaker segmentation and diarization using lexical and acoustic cues via sequence to sequence neural networks. In *INTERSPEECH*.

Park, T. J., Han, K. J., Huang, J., He, X., Zhou, B., Georgiou, P., et al. (2019). Speaker diarization with lexical information. In *Interspeech* (pp. 391–395).

Park, T. J., Han, K. J., Lane, I., & Georgiou, P. (2018). SPEAKER diarization with lexical information. arXiv preprint arXiv:1811.10761.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., et al. (2011). The kaldi speech recognition toolkit. In *IEEE 2011 workshop on automatic speech recognition and understanding CONF*. IEEE Signal Processing Society.

Qin, Z., & Kim, D. (2019). Rethinking softmax with cross-entropy: Neural network classifier as mutual information estimator. arXiv preprint arXiv:1911.10688.

Ruder, S., Vulić, I., & Søgaard, A. (2019). A survey of cross-lingual word embedding models. *Journal of Artificial Intelligence Research*, *65*, 569–631.

Shahid, F., Zameer, A., Mehmood, A., & Raja, M. A. Z. (2020). A novel wavenets long short term memory paradigm for wind power prediction. *Applied Energy*, *269*, Article 115098.

Silnova, A., Brümmer, N., Rohdin, J., Stafylakis, T., & Burget, L. (2020). Probabilistic embeddings for speaker diarization. In *Odyssey*.

Snyder, D., Garcia-Romero, D., McCree, A., Sell, G., Povey, D., & Khudanpur, S. (2018). Spoken language recognition using X-vectors. In *Odyssey* (pp. 105–111).

Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., & Khudanpur, S. (2018). X-vectors: Robust dnn embeddings for speaker recognition. In *ICASSP* (pp. 5329–5333). IEEE.

Sun, Z., Yu, H., Song, X., Liu, R., Yang, Y., & Zhou, D. (2020). Mobilebert: a compact task-agnostic bert for resource-limited devices. arXiv preprint arXiv:2004.02984.

Sunitha, C., & Chandra, E. (2015). Speaker recognition using MFCC and improved weighted vector quantization algorithm. *International Journal of Engineering and Technology (IJET)*, *7*(5), 1685–1692.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems* (pp. 3104–3112).

Tian, Z., Yan, R., Mou, L., Song, Y., Feng, Y., & Zhao, D. (2017). How to make context more useful? an empirical study on context-aware neural conversational models. In *Proceedings of the 55th annual meeting of the association for computational linguistics (Volume 2: short papers)* (pp. 231–236).

Vassilvitskii, S., & Arthur, D. (2006). K-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on discrete algorithms* (pp. 1027–1035).

Vidal, R. (2011). Subspace clustering. *IEEE Signal Processing Magazine*, *28*(2), 52–68.

Wan, L., Wang, Q., Papir, A., & Moreno, I. L. (2018). Generalized end-to-end loss for speaker verification. In *ICASSP* (pp. 4879–4883).

Wang, Y.-Y., Acero, A., & Chelba, C. (2003). Is word error rate a good indicator for spoken language understanding accuracy. In *2003 IEEE workshop on automatic speech recognition and understanding (IEEE Cat. No. 03EX721)* (pp. 577–582). IEEE.

Wang, Q., Downey, C., Wan, L., Mansfield, P. A., & Moreno, I. L. (2018). Speaker diarization with lstm. In *ICASSP* (pp. 5239–5243).

Wang, P., Liu, H., So, A. M.-C., & Balzano, L. (2022). Convergence and recovery guarantees of the K-subspaces method for subspace clustering. In *International conference on machine learning* (pp. 22884–22918). PMLR.

Wang, Y., Zhang, J., Kan, M., Shan, S., & Chen, X. (2020). Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 12275–12284).

Wu, L., Li, S., Hsieh, C.-J., & Sharpnack, J. L. (2019). Stochastic shared embeddings: Data-driven regularization of embedding layers. In *Advances in neural information processing systems* (pp. 24–34).

Xin, J., Nogueira, R., Yu, Y., & Lin, J. (2020). Early exiting BERT for efficient document ranking. In *Proceedings of sustaiNLP: workshop on simple and efficient natural language processing* (pp. 83–88).

Yang, Y., & Xiang, C. (2019). Improve language modelling for code completion through learning general token repetition of source code. In *ICSEKE* (pp. 667–777).

Yang, W., Zuo, W., & Cui, B. (2019). Detecting malicious urls via a keyword-based convolutional gated-recurrent-unit neural network. *IEEE Access*, *7*, 29891–29900.

Yella, S. H., Stolcke, A., & Slaney, M. (2014). Artificial neural network features for speaker diarization. In *2014 IEEE spoken language technology workshop (SLT)* (pp. 402–406). IEEE.

Zajíc, Z., Soutner, D., Hrúz, M., Müller, L., & Radová, V. (2018). Recurrent neural network based speaker change detection from text transcription applied in telephone speaker diarization system. In *International conference on text, speech, and dialogue* (pp. 342–350).

Zhang, Z. (2018). Improved adam optimizer for deep neural networks. In *2018 IEEE/ACM 26th international symposium on quality of service (IWQoS)* (pp. 1–2). IEEE.

Zhang, P., Ouyang, W., Zhang, P., Xue, J., & Zheng, N. (2019). Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 12085–12094).

Zhang, Z., & Sabuncu, M. (2018). Generalized cross entropy loss for training deep neural networks with noisy labels. In *Advances in neural information processing systems* (pp. 8778–8788).

Zhang, A., Wang, Q., Zhu, Z., Paisley, J., & Wang, C. (2019). Fully supervised speaker diarization. In *ICASSP* (pp. 6301–6305).