Ministère de l'enseignement Supérieur et de la recherche Scientifique

وزارة التعليم العالي والبحث العلمي

Badji Mokhtar Annaba University
Université Badji Mokhtar – Annaba
Faculté de Technologie

Département Informatique

جامعة باجي مختار – عنابـــة

كـلي ـة التكـنولوجي ـا

قســم الاعلام الالـي

Thèse

Présentée pour obtenir le diplôme de

Doctorat En-Sciences

Spécialité : Informatique

Par :

BENATI Nadia

Thème :

Détection de mots clés

Thèse soutenue le ….. date de soutenance ……. devant le jury composé de :

N°	Nom et prénom	Grade	Etablissement	Qualité
01	SARI Toufik	Prof.	Université Badji Mokhtar -Annaba	Président
02	**BAHI Halima**	Prof.	Université Badji Mokhtar -Annaba	Rapporteur
03	REZEG Khaled	Prof.	Université Med Khider Biskra	Examinateur
04	SOUICI Labiba	Prof.	Université Badji Mokhtar -Annaba	Examinateur
05	ZARZOUR Hafed	Prof.	Université Med Cherif Messaadia Souk ahras	Examinateur
06	LACHTAR Nadia	MCA	ENSTI - Annaba	Examinateur

DEDICACES

A la mémoire de mes parents

REMERCIEMENTS

Je viens par ces mots rendre hommage à mon amie et encardreur Pr Bahi Halima pour son accompagnement et ses encouragements durant toutes ces années « de traversée de désert ». Je tiens à lui exprimer toute ma considération pour son soutien et son aide, sans sa détermination ce travail n'aurait jamais vu le jour. Qu'elle trouve ici l'expression de ma sincère gratitude et reconnaissance.

Mes vifs remerciements au Pr Sari Toufik pour avoir accepté de présider mon jury de soutenance et pour son aide et sa compréhension.

Je voudrais exprimer ma gratitude et mes remerciements à mon amie Dr Lachtar-Gouasmia Nadia d'avoir accepté d'examiner cette thèse et pour sa présence et ses encouragements durant toutes ces années.

Mes remerciements et ma gratitude vont au Pr Souici-Meslati Labiba pour ses encouragements et son soutien et pour avoir accepteé d'examiner ce travail.

Je tiens à exprimer mes remerciements à mon collègue Pr Rezeg Khaled qui, malgré la distance, a accepté de faire partie du jury.

Mes sincères remerciements à mon collègue Pr Zarzour Hafed d'avoir accepté d'examiner cette thèse malgré ses occupations.

Mes vifs remerciements vont aussi à toute ma famille et mes amies pour leur soutien

Je remercie mes collègues pour leur encouragements.

Je dis merci à toute personne ayant contribué de près ou de loin à la réalisation de ce travail.

Table des Matières

Table de figures

ملخص

أتاح التقدم التكنولوجي وسعات تخزين البيانات الضخمة مجموعات كبيرة من البيانات الصوتية. ويلعب الكلام دورًا بارزًا في هذه البيانات، ومع ذلك، فإن استغلال هذه البيانات يتطلب تطوير أدوات مناسبة، لأن هذه البيانات غالبًا ما تكون بيانات خام غير موسومة ولا مشروحة، وتحتوي على الكثير من الضوضاء من مختلف الطبائع. أحد التطبيقات الممكنة للاستفادة من هذه البيانات هو الكشف عن الكلمات المفتاحية التي يمكن أن تشارك في بناء أنظمة استرجاع المعلومات، أو التحكم الآلي في الأنظمة مثل الأوامر الصوتية. في هذا السياق، يقترح هذا العمل نهجًا غير خاضع للإشراف للكشف عن المصطلحات المنطوقة في دفق صوتي. يتمثل هدفنا في إنشاء نظام آلي قادر على التعرف على كلمات محددة دون الحاجة إلى بيانات مصنفة مسبقًا . وقد انبثق عن هذه المشكلة اقتراحان. الأول هو جزء من النهج الكلاسيكي الذي يعتمد حصرياً على الجانب الصوتي للإشارة. وهنا، في مرحلة الكشف، يتم استخدام خوارزمية حساب المسافة الديناميكية لقياس التشابه بين كل مقطع كلامي من الدفق المراد استكشافه والمصطلح المنطوق المراد اكتشافه. يستفيد الاقتراح الثاني من أساليب جديدة من التعلم العميق. نقترح هنا طريقة جديدة غير خاضعة للإشراف تعتمد على تقنيات معالجة الصور والرؤية الحاسوبية التي تحملها شبكة عصبية تلافيفية، للسماح بالكشف غير الخاضع للإشراف عن الكلمات المراد البحث عنها والتي لا يعرفها النظام مسبقًا. تم إجراء تجارب لتقييم مقترحاتنا على كل من مجموعة الكلام التي اقترحتها الجمعية الدولية للصوتيات (IPA) وعلى مجموعة أوامر الكلام الشهيرة من Google. النتائج التي تم الحصول عليها واعدة ومهدت الطريق لإجراء المزيد من التجارب التي تعالج نقص وضع العلامات والشروح لبيانات الكلام.

الكلمات المفتاحية: اكتشاف المصطلح المنطوق، نهج غير خاضع للإشراف، التعلم العميق، الشبكة العصبية التلافيفية (CNN).

Résumé

Les avancées technologiques et les énormes capacités de stockage de données ont rendu disponible de grands ensemble de données audio. La parole prend une place prépondérante dans ces données, toutefois, l'exploitation de ces données nécessite le développement d'outils adéquats, car ces données sont le plus souvent des données brutes ni étiquetées ni annotées, et qui contiennent beaucoup de bruit, de diverses natures. Une des applications possibles pour tirer profit de ces données est la détection de mots clés qui pourrait participer à la construction de systèmes de recherche d'information, ou de pilotage de système automatique telle que la commande vocale. Dans ce contexte, ce travail propose une approche non supervisée pour la détection de mots clés dans un flux audio. Notre objectif est de créer un système automatique capable de reconnaître des mots-clés spécifiques sans avoir besoin de données préétiquetées. De cette problématique se sont détachées deux propositions,. La première s'inscrit dans l'approche classique qui s'adosse exclusivement sur l'aspect acoutsitque du signal. Ici, dans la phase de détection, un algorithme de calcul dynamique de distance est utilisé pour mesurer la similarité entre chaque segment parole du flux à prospecter et le mot clé à rechercher. La seconde proposition tire profit des nouvelles méthodes issues du deep learning. Ici, nous proposons une nouvelle méthode non supervisée qui se base sur les techniques du traitement d'image et de la vision par ordinateur portée par un réseau de neurones convolutionnel, pour permettre la détection non supervisée de mots à rechercher qui ne sont pas connus d'avance par le système. Des expériences sont menées pour évaluer nos propositions aussi bien sur le corpus parole proposé par l'association internationale de phonétique (IPA) que sur le très célebre Google Speech Command corpus. Les résultats que nous avons obtenu sont prometteurs et ouvrent la voie à d'autres expérimentations qui adressent le manque d'étiquettage et d'annotation des données parole.

Mots clés:Détection de mots clés parlés, approche non supervisée, deep learning, réseau de neurones convolutionnel (CNN).

Abstract

Technological advances and huge data storage capacities have made large sets of audio data available. Speech plays a prominent role in these data, however, the exploitation of these data requires the development of adequate tools, because these data are most often raw data that are neither labeled nor annotated, and that contain a lot of noise of various natures. One of the possible applications to take advantage of these data is the detection of keywords that could participate in the construction of information retrieval systems, or automatic system control such as voice command. In this context, this work proposes an unsupervised approach for the detection of spoken terms in an audio stream. Our objective is to create an automatic system capable of recognizing specific words without the need for pre-labeled data. Two proposals have emerged from this problem. The first is part of the classical approach which relies exclusively on the acoustic aspect of the signal. Here, in the detection phase, a dynamic distance calculation algorithm is used to measure the similarity between each speech segment of the stream to be prospected and the spoken term to detect. The second proposal takes advantage of new methods from deep learning. Here, we propose a new unsupervised method based on image processing and computer vision techniques carried by a convolutional neural network, to allow the unsupervised detection of words to be searched that are not known in advance by the system. Experiments are conducted to evaluate our proposals both on the speech corpus proposed by the International Phonetics Association (IPA) and on the very famous Google Speech Command corpus. The obtained results are promising and paved the way to further experiments that address the lack of labeling and annotation of speech data.

Keywords: Spoken term detection, unsupervised approach, deep learning, convolutional neural network (CNN).

Chapitre 1 : Introduction Générale

1.1. Introduction

La détection de termes parlés (En Anglais Spoken Term Detection STD) est le processus qui permet de détecter automatiquement la présence de mots ou de phrases spécifiques (ou partie de parole) dans un flux continu de parole. Lorsque cette tâche est conduite automatiquement, elle ouvre la porte à une foultitude d'applications ; cette technologie est de plus en plus utilisée dans de nombreux domaines, notamment la reconnaissance vocale, les centres d'appels automatisés et la transcription de la parole en texte. Elle permet aux systèmes de répondre à des commandes vocales, d'identifier des types de conversations spécifiques et de générer des transcriptions précises du contenu parlé. La STD est un élément clé de nombreuses applications modernes de la parole et a le potentiel de transformer la façon dont nous interagissons avec les ordinateurs et les appareils électroniques en général. On parle aussi de détection de mots clés (keyword spotting) lorsqu'il s'agit de retrouver des mots particuliers.

Actuellement, le keyword spotting(KWS) repose sur des techniques avancées de traitement du signal et de l'apprentissage automatique pour identifier les phonèmes tel que les mots et les phrases dans le flux audio.Les techniques de traitement de signal sont utilisées pour extraire des caractéristiques des signaux audio, telles que la fréquence et la puissance, afin de permettre la détection de mots-clés ou de phrases prédéfinis. Les techniques d'apprentissage automatique sont souvent utilisées pour entraîner des modèles de reconnaissance vocale capables de reconnaître les mots et les phrases qui constituent les termes recherchés. Ces modèles sont entraînés sur des données d'apprentissage contenant des enregistrements audios de différents locuteurs et de différentes conditions acoustiques, afin d'apprendre à reconnaître les différentes variations du langage parlé.

Il est toute fois important de mentionner que de tels systèmes sont difficiles, voire impossible, à implémenter pour toutes les langues, car leur mise en œuvre nécessite l'existence d'une quantité immense de données étiquetées. Les approches de la détection non supervisé nous semblent être une alternative dans ce cas. Dans un système de détection de mots clés non supervisé, des techniques de traitement du signal et

d'apprentissage automatique sont utilisées pour extraire des caractéristiques significatives du flux audio. Les segments acoustiques partageant des similitudes acoustiques sont ensuite regroupés à l'aide de techniques de clustering ou de classification non supervisée. Les méthodes non supervisées pour le KWS peuvent être particulièrement utiles lorsqu'il n'y a pas suffisamment de données étiquetées pour entraîner un modèle supervisé. Les avantages de l'approche non supervisée incluent également la capacité à s'adapter à de nouveaux environnements acoustiques et à identifier des termes inattendus.

1.2. Contexte et Motivation

À l'ère du Big Data et de la reconnaissance vocale, une énorme quantité de données audio est disponible. Par conséquent, le traitement du signal est devenu très important pour tirer le meilleur parti de ces données et résoudre de nombreux problèmes dans divers domaines, y compris l'amélioration de l'expérience utilisateur, la sécurité,la surveillance des médias sociaux, la gestion des centres d'appel, la surveillance de la qualité, etc.

D’autre part, le recours à la détection non supervisé de mots parlés, nous semble une problématique intéressante à aborder dans ce contexte vu que le contenu audio malgré sa profusion est difficile à annoter et à étiqueter. En effet,les succès actuels des méthodes non supervisées nous encouragent à investiguer des solutions issues de cette orientation.

Nous nous positionnons dans la recherche de mots clés dans un flux de parole, qui peut être appliquée dans de nombreux domaines.

Figure 1.1: Interaction homme-machine via les signaux audios

Pour répondre au problème de la détection de mots parlés selon une approche non

supervisée, nous proposons deux approches. La première est exclusivement acoustique, et se base sur des mesures physique telles que l'énergie du signal. La seconde proposition s'adosse aux nouvelles tendances de l'intelligence artificelle que sont les algorithmes du deeplearning, en particulier, nous proposons l'utilisation d'un réseau de neurones convolutionnel entrainé dans un contexte auto-supervisé (self-supervision).

1.3. Organisation de la thèse

Cette thèse présente le travail qui consiste à proposer un système pour la détection des mots parlés dans un flux de parole selon une approche non supervisée. La thèse est organisée comme suit :

- Le chapitre 1 introduit le contexte de notre travail qui est la détection des mots parlés,ainsi que nos motivations pour entreprendre ce travail.
- Le chapitre 2 est une revue de la littérature sur la détection des mots parlés, incluant les différentes approches et techniques existantes issues de l'approche non supervisée. Le chapitre introduit, d'abord, la détction de mots clés (KWS), ensuite on y vient à la détection de mots parlés. Ensuite, le chapitre présente une revue de la littérature des méthodes qui existent concernant la détection des mots parlés.
- Le chapitre 3 introduit l'apprentissage profond, ainsi que son utilisation pour la détection de mots parlés.
- Le chapitre 4 présente la première contribution dans le cadre de ce travail, à savoir la proposition d'une méthode exclusivement basée sur les méthodes acoustiques.
- Dans le chapitre 5 nous présentons notre seconde proposition, qui s'appuie sur les nouvelles tendances issues de l'apprentissage profond, on y présente aussi les résultats obtenus ainsi que leur analyse et discussion.
- Cette thèse se termine par une conclusion et des perspectives au travail réalisé.

Chapitre 2 :

Détection de mots clés

2.1. Introduction

Les techniques de recherche audio consistent généralement à récupérer un fichier audio souhaité, à partir d'une base de données audio, en donnant un mot-clé ou une requête parlée au système.

Les données audio comprennent tout les sons audibles dans la gamme de fréquences de 20 à 20 000 Hz, comme les données vocales, la musique, les sons d'animaux, les sons de klaxons, les rires, les gazouillis d'oiseaux, les archives d'actualités et les conférences audio (Leena et Deekshitha, 2019). Les premiers systèmes de recherche ne pouvaient fonctionner qu'avec des mots-clés sous forme de texte. Mais les systèmes récents ont la capacité d'effectuer des recherches à partir d'une requête audio/mot-clé parlé.

L'objectif principal de la détection de mots clés (KWS pour keyword spotting) et des termes parlés (STD pour spoken term détection) est de repérer des mots-clés ou des séquences de mots dans les occurrences de signaux d'entrée (Wang, 2010). La détection de mots-clés est considérée comme la première méthode de recherche audio. Par la suite, la détection de mots-clés a ouvert la voie à la détection de termes parlés (STD) et à la détection de requêtes par l'exemple (QbE-STD pour query by example STD). La requête est généralement un mot-clé ou une phrase courte qui est donnée au système pour récupérer un fichier audio contenant cette requête. A ce titre, un mot-clé doit être suffisamment fort pour décrire de manière unique le fichier souhaité dans la base de données.

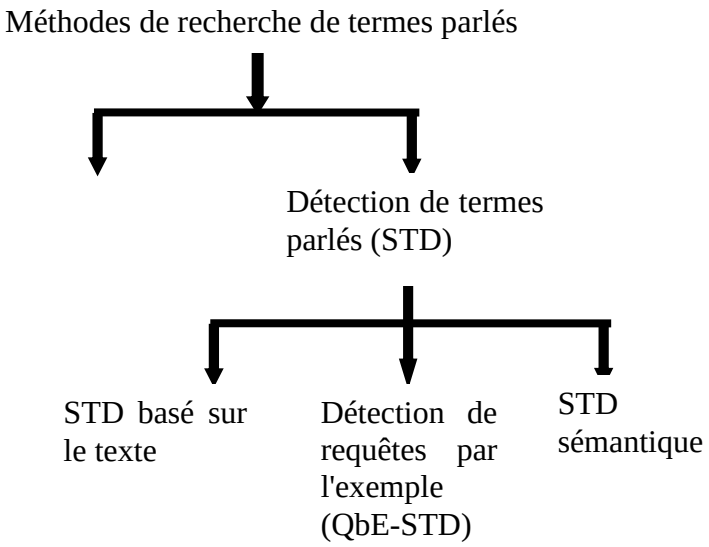


Figure 2.01 : Organigramme structurel des techniques de détection de mots parlés

2.2. Détection de mots-clés (KWS)

Le système est conçu pour un ensemble de mots-clés prédéfinis. Les mots-clés sont sélectionnés de manière à ce que les mots/phrases apparaissent plus fréquemment dans la base de données. La détection peut être effectuée de manière supervisée ou non supervisée. Dans la technique KWS, la conversion de la parole en texte est effectuée pour la requête et pour la base de données. La recherche au niveau du texte est effectuée ultérieurement. Un système de reconnaissance de la parole continue à large vocabulaire(LVCSR pour large vocabulary continuous speech recognizer) ou ses variantes sont généralement utilisés pour la conversion de la parole en texte. Cette méthode n'est donc utile que pour les langues bien dotées en ressources.

2.2.1. Applications des systèmes de KWS

Les principales applications des KWS sont la surveillance des mots-clés, l'indexation des documents audio, les dispositifs de contrôle des commandes et les systèmes de dialogue. Les applications spéciales de surveillance des mots-clés sont l'écoute téléphonique, la surveillance des dispositifs d'écoute et la surveillance des émissions.(Thambiratnam 2005)

- Indexation de documents audio : Un document audio fait l'objet d'une recherche rapide de mots-clés intéressants. Il s'agit d'un moteur de recherche

similaire à un moteur de recherche textuel tel que Google. Cependant, il fonctionne sur des documents audio et non sur des archives textuelles.

- Dispositifs de contrôle des commandes : Ils contrôlent le flux audio et agissent lorsqu'une commande spécifique est détectée. Les téléphones mobiles à commande vocale, les machines industrielles à commande vocale et à commande, les jeux informatiques, les guichets automatiques, l'aide aux personnes handicapées, les formulaires en ligne, l'utilisation de commandes vocales dans la conduite, la commande d'appareils et de logiciels par la parole sont autant d'exemples de dispositifs à commande vocale.

- Systèmes de dialogue : Ils sont généralement exploités dans les environnements commerciaux en remplacement des centres d'appels gérés par des humains.

2.2.2. Architecture d'un système KWS

L'architecture classique d'un système de détection de mots clés est schématisée dans la figure 2.2. Il inclut les étapes de prétraitment, d'extraction de caractéristiques, et de détection selon l'approche adoptée.

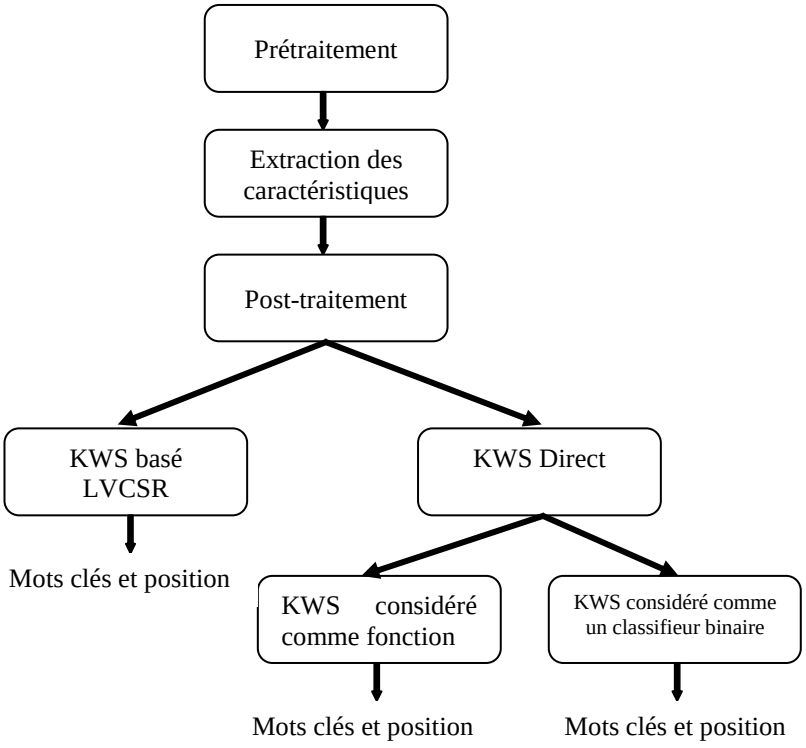


Figure 2.2 : Architecture classique d'un système de KWS

La phase de prétraitement peut inclure la détection de la présence ou non de la parole, ou parfois seulement l'élimination du silence dans le signal audio d'entrée, il peut s'agir aussi d'éliminer les bruits qui nuisent à la bonne conduite du processus de détection de mots, tels que les bruits de fond.

L'opération suivante est l'extraction de caractéristiques. La performance des systèmes de traitement de la parole diminue considérablement lorsque la parole est corrompue par le bruit de fond ou la distorsion du canal. Bien qu'il existe des méthodes d'extraction de caractéristiques résistantes à ces types de bruit (Huang et al., 2001 ; Li et al., 2014), les caractéristiques les plus courantes, telles que les coefficients cepstraux de fréquence Mel (MFCC), ne sont pas résistantes aux bruits additifs ou convolutifs. Pour surmonter ce problème, les méthodes de traitement de la compensation environnementale normalisent la structure temporelle ou spectrale des caractéristiques. La structure temporelle des caractéristiques peut être normalisée à l'aide de différentes méthodes d'amélioration de la parole (Ngo et al., 2012 ; Tabibian et al., 2015 ; Vaseghi, 2008). Afin de normaliser la structure spectrale des caractéristiques, diverses techniques ont été proposées, telles que la normalisation de la moyenne cepstrale (CMN), la normalisation de la variance cepstrale (CVN) (Viikki et al., 1998), la normalisation de la moyenne et de la variance cepstrales (CMVN), la normalisation du gain cepstral (CGN) (Yoshizawa et al., 2004), les techniques d'égalisation d'histogramme (HEQ) (Molau et al., 2003), le traitement spectral relatif (RASTA) (Hermansky et Morgan, 1994 ; Hermansky et al., 1991), et le filtrage ARMA (Chen et Bilmes, 2007).

D'autre part, il existe deux approches pour concevoir des systèmes KWS. Dans la première approche, appelée "KWS basé sur la LVCSR", un système de reconnaissance vocale à grand vocabulaire (LVCSR) convertit l'énoncé vocal d'entrée en texte, puis un algorithme de recherche textuelle détermine les mots-clés présents dans l'énoncé (Cernocky et al., 2007 ; Miller et al., 2007 ; Motlicek et al., 2012 ; Ramabhadran et al., 2009 ; Szöke et al., 2005b ; Wang et al., 2012 ; Weintraub, 1995). Le KWS basé sur la LVCSR se compose de deux phases. Tout d'abord, un reconnaisseur de parole à large vocabulaire convertit de grandes archives audio en treillis de phonèmes ou de mots. Ensuite, dans la deuxième phase, la recherche basée sur les treillis recherche l'ensemble des mots-clés cibles. La première phase du KWS basé sur la LVCSR est hors ligne, tandis que la seconde est en ligne. Cette approche présente trois inconvénients majeurs (Tabibian,

2020). Premièrement, une grande quantité de données étiquetées est nécessaire pour former les KWS basés sur les LVCSR. Deuxièmement, le coût de calcul lié au décodage d'un vocabulaire étendu est élevé. Le troisième inconvénient est la diminution de ses performances pour les mots hors vocabulaire (OOV). Différentes méthodes ont été proposées pour résoudre le problème des mots hors vocabulaire (Bazzi, 2002 ; Burget et al., 2008 ; Lin et al., 2007 ; Rastrow et al., 2009 ; Szöke, 2010).

Dans la seconde approche appelée "KWS direct", le KWS est considéré comme une tâche de classification sans passer par l'étape de la reconnaissance vocale. Dans cette approche, le KWS est complètement indépendant de la tâche de reconnaissance vocale. La manière la plus intuitive de rechercher des énoncés parlés est de rechercher directement les parties des énoncés qui ressemblent aux mots-clés cibles. Une ou plusieurs occurrences des mots-clés sont utilisées comme modèles, puis comparées au signal audio d'entrée pour prendre des décisions sur les occurrences des mots-clés. L'approche la plus largement utilisée pour la mise en correspondance des modèles est la déformation temporelle dynamique (DTW) (Bridle, 1973 ; Zhang et Glass, 2009). Le KWS basé sur une requête par l'exemple (QBE) est la version améliorée du KWS basé sur DTW qui représente chaque mot comme un vecteur et calcule uniquement les similitudes entre deux vecteurs.

Le détecteur de mots-clés peut être considéré comme une fonction appelée "KWS en tant que fonction" ou comme un classifieur binaire. Dans le premier groupe Ahmad et al., 2009 ; Bahi et Benati, 2009 ; Keshet et Bengio, 2009 ; Tabibian et al., 2011 ; Thambiratnam, 2005), le détecteur de mots clés est une fonction avec deux arguments d'entrée et deux arguments de sortie. Les arguments d'entrée sont les signal audio d'entrée et l'ensemble des mots-clés cibles. L'une des nombreuses méthodes peut être utilisée pour calculer la mesure de confiance de l'occurrence des mots-clés dans le signal d'entrée (Benayed et al., 2003 ; Ferrer et Estienne, 2001 ; Lee et al., 2004;Ou et Luo, 2012;Wang et al., 2009). Si la mesure de confiance calculée est supérieure à un seuil prédéfini, le détecteur de mots clés confirme l'occurrence du mot clé cible dans la position correspondante du signal audio d'entrée.

Dans le second groupe, le keyword spotter est un classifieur binaire qui sépare la classe des phrases contenant des mots-clés cibles de la classe des phrases sans mots-clés cibles (Ayed et al., 2002; Keshet et al., 2009 ; Tabibian et al., 2013). Chaque méthode de classification se compose de deux parties importantes : l'extraction des caractéristiques et

la classification. Dans la partie extraction des caractéristiques, certaines caractéristiques discriminantes sont extraites des signaux audio d'entrée. Ces caractéristiques modélisent la mesure de confiance de l'occurrence du mot-clé cible dans le signal d'entrée et sa position. Dans la partie classification, un classifieur est utilisé pour séparer les deux classes mentionnées avec un taux d'erreur minimal, selon une mesure d'évaluation.

2.3. La détection de termes parlés

Dans ce cas, la requête est acquise sous la forme d'un terme parlé ou d'un fichier audio segmenté. Les systèmes de détection de termes parlés (STD) peuvent être classés comme suit : STD basé sur le texte, détection de termes parlés par l'exemple (QbE-STD) et interrogation sémantique par l'exemple (Semantic-QbE).

Dans la STD basée sur le texte, la requête audio est d'abord convertie en texte/symboles correspondants. Ensuite, une recherche textuelle est effectuée pour trouver l'occurrence du mot à rechercher dans la base de données. Les systèmes de reconnaissance automatique de la parole (ASR pour automatic speech recognition) sont nécessaires pour la conversion de la parole en texte. Comme les systèmes ASR ont besoin d'une grande quantité de données audio annotées pour leur développement, ils ne peuvent pas être utilisés pour les langues insuffisamment dotées en ressources.

Dans la STD Query-By-Example (QbE), la requête est traitée directement. Il n'y a pas de conversion de la parole en texte dans QbE-STD. L'utilisateur présente au système les extraits audio souhaités contenant des requêtes. Le système recherche alors dans la base de données les segments qui ressemblent le plus à ces requêtes. La plupart des systèmes QbE utilisent des méthodes de correspondance de modèles (template matching) comme DTW ou ses variantes.

La Semantic-QbE(wake-up word detection WUW) est similaire à la KWS, sauf qu'elle est capable de détecter uniquement les mots/phrases utilisés dans des contextes spécifiques.

Table 2.1. KWS vs STD

Approche	Remarques
KWS	Ne peut récupérer que des mots-clés prédéfinis ; supervisé
STD	Open KWS ; pas de connaissance préalable du mot-clé ; utilise le système LVCSR ; la conversion de la parole en symboles est nécessaire, le système dépendra de la langue ; généralement supervisé
QbE-STD	Pas besoin de conversion de la parole en symboles, possibilité de réaliser des STD multiligues ; généralement non supervisé.
QbE sémantique	Peut retrouver l'emplacement d'une requête sémantique, les segments de parole sont mis en correspondance avec les vecteurs d'intégration par couplage avec le contexte visuel ; non supervisé ; peut traiter les langues à faibles ressources.

2.3.1. Les techniques de développement des systèmes STD

La détection de mots-clés fait référence à la recherche et l'extraction d'un ensemble fermé de mots-clés. La technologie s'est aujourd'hui améliorée et permet de rechercher et d'extraire avec une grande précision n'importe quel mot parlé non spécifié (spoken term) d'une base de données audio. C'est ce que l'on appelle la détection de termes parlés (STD). Les systèmes de STD peuvent être classés en deux catégories : les systèmes de STD basés sur le texte ou simplement STD et les systèmes de STD basés sur l'interrogation par l'exemple (QbE-STD). En fonction de la manière dont ils sont réalisés, les systèmes de STD sont classés en deux catégories : supervisés et non supervisés. Il est à remarquer que dans ce qui suit, nous utilisons parfois le séquence « mot clé » au lieu de mot parlé, qui est la traduction de l'anglicisme « spoken word » car elle nous semble plus facile à appréhender, il s'agit toutefois ici de mots qui ne sont pas connu à l'avance par le système, mais introduit par l'utilisateur lors de sa requête.

2.3.1.1. Détection de termes parlés à l'aide de l'apprentissage supervisé

A. Modèles de postériogrammes phonétiques

Le système QbE-STD illustré à la figure 2.3 propose la réalisation de QbE-STD à l'aide de

postériogrammes phonétiques (Hazenet al., 2009). Un système de reconnaissance phonétique développé à l'université de technologie de Brno (BUT) (Schwarz et al., 2003) est utilisé pour générer des postériogrammes phonétiques pour les occurrences de la requête et du test. La similarité entre ces deux postériogrammes est ensuite calculée. Les postériogrammes de la requête et du test sont comparés et une matrice de similarité n × m est obtenue en calculant la similarité entre la distribution à posteriori de tous les n segments de la requête et celle de tous les m segments du test. L'algorithme DTW modifié est ensuite utilisé pour trouver un chemin bien adapté entre la requête et le segment de test. Il existe deux approches pour utiliser efficacement les multiples occurrences de requêtes disponibles. La première consiste à combiner tous les modèles en un seul modèle afin qu'il puisse refléter les caractéristiques de tous les exemples de requêtes. La deuxième approche, qui est coûteuse en temps de calcul, consiste à utiliser toutes les requêtes disponibles pour générer des scores. Ces scores sont combinés par fusion/moyennage des scores.

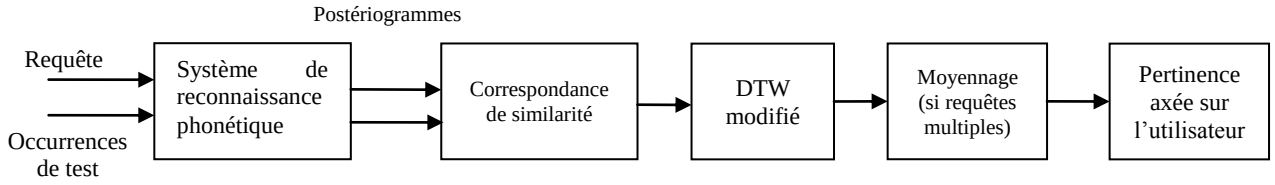


Figure 2.3 : Diagramme pour QbE STD utilisant des modèles phonétiques de postériogrammes

B. Sac de mots acoustiques basé sur les segments

La figure 2.4 présente la méthode de QbE-STD utilisant la technique du sac de mots acoustiques (BoAW pour bag of acoustic words) basée sur les segments. Après l'extraction des caractéristiques, les mots acoustiques significatifs sont identifiés et les emplacements des mots clés dans la base de données sont déterminés. Un score d'histogramme est calculé pour chaque segment de la base de données. Il s'agit du nombre de fois où ce segment est extrait par les mots acoustiques significatifs du segment de la requête. Plus le score d'histogramme est élevé, plus la probabilité que le segment corresponde à la requête est grande. L'algorithme DTW est effectué entre les postériogrammes gaussiens de la requête

et ceux des segments de la base de données les plus probables. Ainsi, le classement des segments de la base de données est effectué à l'aide du score de l'histogramme BoAW et du score DTW. Le score fusionné du segment de base de données est calculé. Les emplacements des mots clés ont les valeurs les plus faibles du score combiné, les segments de la base de données sont donc classés par ordre croissant de leur score combiné.

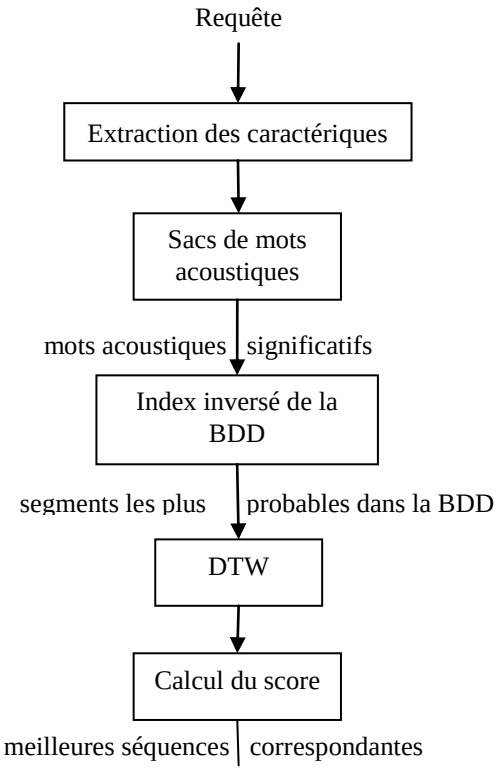


Figure 2.4 :Diagramme pour QbE STD utilisant les sacs de mots acoustiques (BoAW)

C. Décodage phonétique

Dans Wallance at al. (2007), un décodeur phonétique et la technique Dynamic Match Lattice Spotting (DMLS) sont utilisés pour trouver rapidement les termes recherchés. Le système comporte principalement deux étapes, comme le montre la figure 2.5. Dans la phase d'indexation, la base de données est indexée et organisée après le décodage phonétique. A la phase recherche, le mot-clé est converti en représentation phonétique correspondante et le DMLS est utilisé pour localiser les emplacements similaires (Thambiratnam et Sridharan, 2007). Si le mot-clé est un mot hors vocabulaire, les règles

lettre-son (graphème/phonème) sont utilisées pour obtenir les prononciations phonétiques correspondantes. Si les phonèmes sont mis en correspondance avec les classes correspondantes (voyelles, nasales, etc.), ces hyper-séquences sont utilisées pour la mise en correspondance dynamique (dynamic matching). Cela permet de réduire l'espace et le temps de recherche. La distance minimale d'édition (MED) est utilisée pour localiser une séquence correspondante dans la base de données. Le score d'occurrence du mot-clé est estimé en fusionnant linéairement le score MED avec le score du rapport de vraisemblance acoustique, ALLR(P). La fusion du ALLR permet de différencier les occurrences ayant le même score MED et de promouvoir l'occurrence ayant une plus grande probabilité acoustique. Étant donné que le score MED n'est pas comparable pour les occurrences ayant des longueurs de phonèmes différentes, une étape de vérification basée sur les réseaux de neurones est utilisée pour produire un score de confiance de détection final pour chaque occurrence de terme.

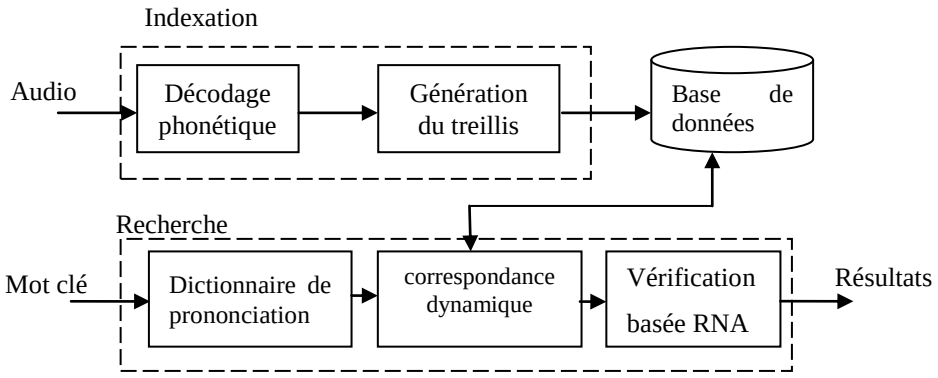


Figure 2.5 : Diagramme pour STD utilisant l'approche phonétique

Étant donné que la recherche dans la STD basée sur les treillis phonétiques est plus lente, Pinto et al. (2008) ont proposé une méthode plus rapide avec une taille d'index plus petite. Pour atteindre l'objectif d'un index réduit et d'une vitesse de recherche plus rapide, seule la meilleure séquence de phonèmes du treillis est retenue. Un modèle de prononciation probabiliste est ensuite proposé pour compenser les erreurs lors de la reconnaissance des phonèmes. La STD utilisant un décodage phonétique rapide est proposée à l'aide de la technique DMLS (Wallance at al. 2007 ; Wallance et al. 2009), par l'utilisation du décodage monophonique en boucle ouverte et à une recherche hiérarchique rapide dans le treillis phonétique. La recherche préliminaire est effectuée à l'aide de la

base de données d'hyper-séquences (HSDB), afin de réduire la vitesse et l'espace de recherche. Ensuite, la deuxième recherche basée sur MED est effectuée à l'aide de la base de données de séquences réelles (SDB) correspondant aux emplacements présélectionnés lors de la recherche préliminaire. Une méthode basée sur l'alignement conjoint des treillis phonétiques générés à partir de la requête et de la base de recherche a montré qu'il est préférable d'utiliser plus d'une chaîne phonétique optimale pour l'énoncé ou la requête (Lin et al. 2008). Un système de recherche et d'indexation basé sur un transducteur d'états finis pondéré (WFST) est proposé, qui permet d'utiliser la représentation du treillis de l'échantillon audio directement comme requête pour effectuer la recherche (Parada et al., 2009). De même, la combinaison des réseaux de confusion de mots et de phonèmes est utile pour l'extraction de mots-clés parlés à vocabulaire ouvert (Hori et al., 2007). L'utilisation d'un réseau de confusion permet d'obtenir une table d'indexation plus compacte afin d'obtenir une correspondance de mots parlés plus robuste par rapport aux techniques typiques basées sur le treillis.

2.3.1.2. Détection de termes parlés à l'aide de l'apprentissage non supervisé

A. Postériogrammes gaussiens

Les postériogrammes gaussiens dans un cadre non supervisé pour l'identification des mots clés ont été utilisés dans (Zhang et Glass 2009 ; Dumpala et al. 2015). Les données d'apprentissage sont d'abord classées en segments vocaux ou non vocaux. Les segments vocaux sont ensuite utilisés pour entraîner le GMM. Les variantes DTW sont utilisées pour comparer les postériogrammes gaussiens des échantillons de mots clés avec ceux des occurrences de test . Les résultats sont ensuite classés sur la base des scores de distorsion.

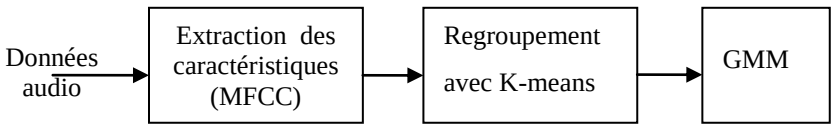


Figure 2.6 : Diagramme pour la modélisation de mélange de gaussiennes

La QbE-STD peut être réalisé en utilisant la DTW segmentaire (S-DTW) des postériogrammes gaussiens, comme le montre la figure 2.7. Un GMM est créé et les postériogrammes gaussiens des données de requête et de test sont calculés à l'aide du

GMM (Dumpala et al. 2015). Ces deux postériogrammes sont alignés à l'aide de l'algorithme S-DTW. Une matrice de distance est construite à l'aide de la métrique du logarithme négatif de la magnitude. Ensuite, la matrice de similarité est dérivée de cette matrice de distance à l'aide de l'algorithme S-DTW et les occurrences possibles des requêtes sont localisées à partir de l'occurrence de test. L'algorithme S-DTW peut être utilisé pour QbE-STD avec deux contraintes (Zhang et Glass 2009). La première est l'ajustement de la condition de fenêtre, qui empêche le chemin de déformation d'aller trop loin en avant ou en arrière dans l'un ou l'autre des postériogrammes. La deuxième contrainte consiste à appliquer différentes coordonnées de départ au processus de déformation, de manière à diviser la matrice de différence en différentes régions diagonales. Le chevauchement des fenêtres coulissantes permet d'éviter les redondances de calcul. Enfin, une région de l'occurrence de test avec un score de distorsion minimal est choisie pour une requête particulière.

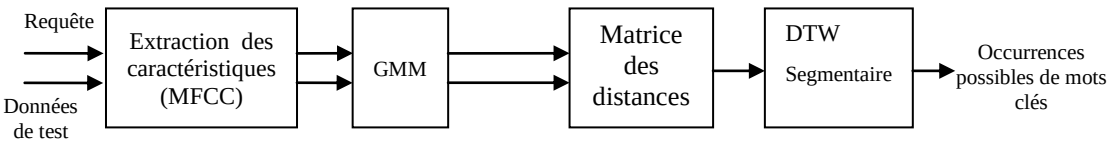


Figure 2.7: Diagramme pour STD utilsant les postériogrammes gaussiens

Dans Mantena et al. (2014), le QbE-STD est traité à l'aide de postériogrammes gaussiens obtenus à partir de plusieurs caractéristiques spectrales et temporelles de la parole. Les caractéristiques spectrales et temporelles extraites du signal vocal sont utilisées pour obtenir les postériogrammes gaussiens correspondants. Les données d'interrogation et de recherche sont converties en postériogrammes correspondants. La recherche et l'extraction sont effectuées à l'aide de variantes de DTW telles que NS-DTW et NS-DTW rapide , comme le montre la figure 2.8.

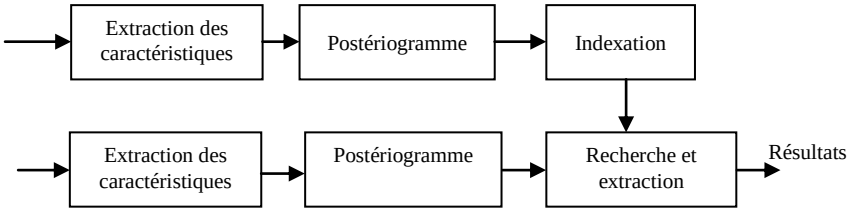


Figure 2.8 : Diagramme pour QbE STD utilsant FDLP(Prédiction linéaire dans le domaine des fréquences) et NS-DTW

Les coefficients cepstraux de Fourier-Bessel sont utilisés pour former un GMM afin d'obtenir une représentation gaussienne du postériogramme (Vasudev et al. 2015). La figure 2.9 montre comment une technique non supervisée est utilisée pour la QbE-STD. Au cours de la phase d'apprentissage, les composantes du GM sont initialisées à l'aide de la méthode de regroupement k-means. Le regroupement sera initialisé en trouvant la moyenne de l'ensemble des données d'apprentissage. Pour obtenir les GP, un vecteur de probabilité de ligne est généré pour chaque fenêtre audio, et la normalisation de la moyenne zéro est également employée. Enfin, pour trouver les emplacements possibles du mot-clé dans les occurrences de test, la méthode S-DTW est utilisée.

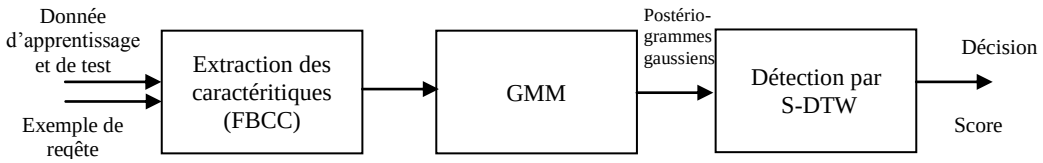


Figure 2.9 : Diagramme pour QbE STD utilisant les caractéristiques de Bessel

B. La modélisation des segments acoustique par les postériogrammes

La technique de modélisation des segments acoustiques (ASM) utilise des postériogrammes pour effectuer une QbE-STD non supervisée (Wang et al. 2011). Les données de requête et de recherche vocales sont converties en postériogrammes ASM correspondants. Ensuite, les occurrences possibles de la requête sont localisées à l'aide de la technique de mise en correspondance des modèles(template matching) basée sur les postériogrammes ASM.

La DTW segmentaire est utilisé dans Wang et al. (2011) pour faire correspondre les postériogrammes ASM. Obara et al. (2017) proposent deux méthodes pour améliorer la vitesse de QbE en utilisant les postériogrammes du DNN. Ces méthodes consistent soit à transformer le postériogrammes en une matrice de bits, soit à utiliser la méthode des vecteurs creux pour remplacer les éléments les moins probables par 0. Ram et al. (2018) présentent un système plus rapide et performant qui dépend de la modélisation de sous-espaces creux pour réduire la dimensionnalité.

C. Segmentation basée sur les délais des groupes (group delay)

Dans cette approche, les données audio sont segmentées en unités acoustiques plus petites

à l'aide de fonctions de retard de groupe (GD pour group delay) (Mantena et al. 2014). Les limites de la segmentation peuvent varier des phonèmes aux mots en utilisant un paramètre appelé windowscale factor dans l'algorithme GD-based Segmentation (GDS).Cette méthode de segmentation présente les avantages suivants :

- Aucune connaissance préalable requise
- Algorithme rapide
- Opérations transparentes sur les mots hors vocabulaire
- Approche indépendante de la langue et du texte

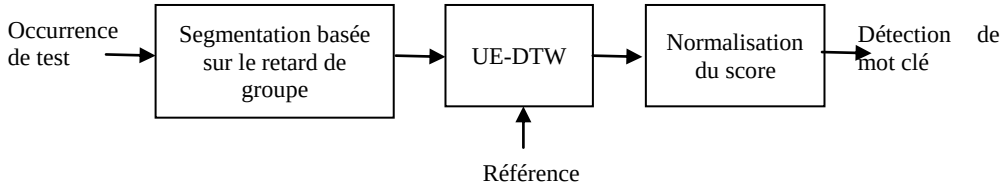


Figure 2.10 : Diagramme pour KWS utilsant la segmentation basée sur le retard de groupe

La détection des mots-clés effectuée à l'aide de la segmentation basée sur le retard de groupe est illustrée à la figure 2.10 (Madikeri et Murthy, 2012). Les limites prédites par l'algorithme de segmentation peuvent contenir des désalignements. Les limites seront donc élargies de quelques fenêtres à chaque extrémité.

Pour trouver l'emplacement des mots-clés, chaque segment est soumis à l'algorithme de correspondance des modèles basé sur la DTW. Les mots-clés sont localisés si la mesure de la distance est inférieure au seuil, sinon le segment est rejeté. Dans Madikeri et Murthy (2012), l'algorithme UE-DTW est utilisé pour supprimer les contraintes liées aux points d'extrémité. La normalisation des scores est effectuée pour réduire le taux de fausses alarmes.

Une QbE-STD rapide utilisant l'algorithme de segmentation basé sur le retard de groupe (Pandia et al., 2016) utilise une stratégie en deux étapes pour repérer les mots-clés avec les MFCC comme vecteurs de caractéristiques. Comme le montre la figure 2.11, lors de la première étape, certaines correspondances du fichier de recherche sont identifiées et sont utilisées comme nouveaux modèles de requête pour la deuxième étape, en plus des modèles de requête initiaux. Trois modèles de mots-clés choisis manuellement sont utilisés comme entrée dans la première passe. L'algorithme GDS est utilisé pour obtenir les

segments au niveau-syllabe du fichier de recherche et des fichiers de requête.

Pendant la première passe, les modèles de requête sont recherchés à chaque limite de syllabe du fichier de recherche. UE-DTW est utilisé pour trouver de nouveaux modèles correspondants. Ainsi, pour les trois modèles de requête d'entrée, trois nouveaux modèles sont obtenus. Ces six modèles sont donc utilisés dans la deuxième passe pour effectuer la S-DTW sur le fichier de recherche.En utilisant la moyenne géométrique de tous les scores DTW, une bonne correspondance du mot-clé est localisée dans le fichier de recherche. Les requêtes utilisées étant de longueurs différentes, le score DTW est calculé et normalisé en fonction du nombre de syllabes de la requête.

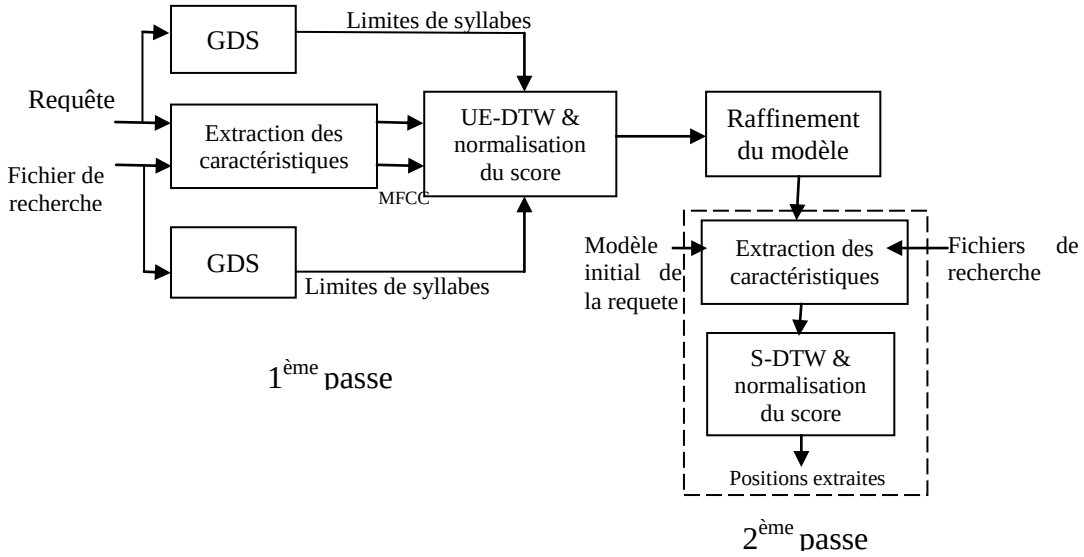


Figure 2.11 : Diagramme pour QbE STD proposé pour les langages à zéro ressource

D. Techniques de traitement morphologique des images

Des techniques de traitement morphologique des images ont été proposées pour détecter les mots-clés à partir d'un signal vocal continu (Sankar et al., 2016). Les données vocales sont converties en postériogrammes de phonèmes correspondants à l'aide d'un système hybride HMM-ANN. Les postériogrrammes basés sur l'ANN donnent une représentation lisse du signal vocal. Les MFCC contextuels sont utilisés comme vecteur de caractéristiques en fusionnant les 4 fenêtres voisines de gauche et de droite ensemble. Une variante de DTW est appliquée aux postériogrammes du mot-clé et de la séquence de test.

La matrice de distance DTW est ensuite représentée sous la forme d'une image en niveaux de gris, puis traitée pour en extraire la ligne diagonale indiquant la présence du mot-clé. L'ensemble des opérations effectuées sur l'image DTW comprend la segmentation, l'inversion, la dilatation, la squelettisation et l'érosion. La segmentation dynamique basée sur le seuil est effectuée pour convertir la matrice d'accumulation des niveaux de gris en image binaire. L'inversion permet de faire passer les objets souhaités de l'arrière-plan au premier plan. La dilatation à l'aide d'un élément structurant permet de réparer les connexions brisées lors de la segmentation. L'image dilatée est puis squelettisée pour obtenir une seule épaisseur du pixel dans l'image binaire. L'érosion permet de soustraire les fausses branches horizontales/verticales de l'image squelettisée. Enfin, l'analyse des composantes connectées permet d'identifier la présence d'un mot-clé dans le flux de parole continu. Le mot-clé est identifié si l'objet au premier plan en contient une partie dans la dernière ligne de l'image

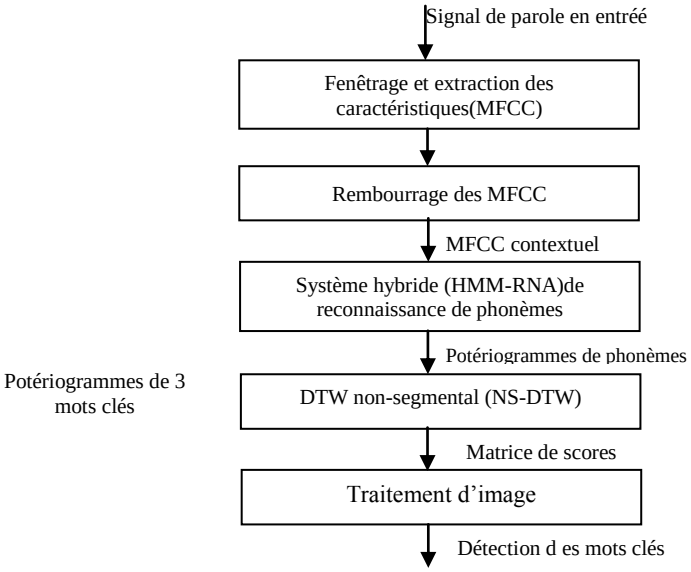


Figure 2.12. :Diagramme pour STD utilsant les techniques de traitement morphologique des images

Chapitre 3 : La détection des mots clés parlés par l'apprentissage profond

3.1. L'apprentissage profond

L'architecture générale d'un système de détection de mots-clés parlés basé sur l'apprentissage profondest composé de trois blocs principaux: 1) l'extracteur de caractéristiques acoustiques qui convertit le signal d'entrée en une représentation compacte de la parole, 2) le modèle acoustique basé sur l'apprentissage profond qui produit des probabilités à postériori pour les différentes classes de mots clés et de mots non-clés à partir des caractéristiques acoustiques du signal, et 3) le gestionnaire de probabilités qui permet de traiter la séquence temporelle des probabilités à postériori afin de déterminer l'existence éventuelle de mots clés dans le signal d'entrée. (Prabhavalkar et al., 2015 ; Chen et al., 2014 ;Sainath et Parada, 2015 ;Pedroni et al., 2018 ; Liu et al., 2019 ; Sørensen et al., 2020)

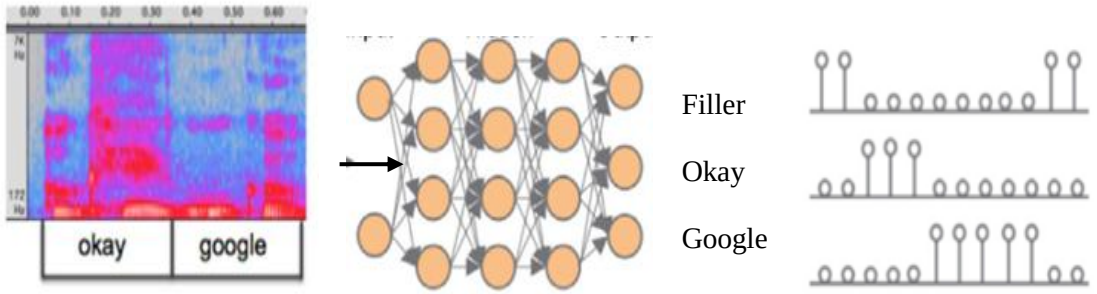


Figure 3.1. Structure d'un système de STD (d'après Chen et al., 2014)

Si nous considérons x(m) comme un signal acoustique comprenant de la parole. Dans un premier temps, l'extracteur de caractéristiques de la parole calcule une représentation alternative de x(m), à savoir X. Il est souhaitable que X soit compact (c'est-à-dire de dimension inférieure, afin de limiter la complexité de calcul), discriminant en termes de

contenu phonétique et robuste aux variations acoustiques López-Espejo (2017). Les caractéristiques vocales X sont traditionnellement représentées par une matrice bidimensionnelle composée d'une séquence temporelle de vecteurs de caractéristiques à K dimensions x_t (t = 0, ..., T - 1) comme dans l'équation :

$$X = (x_0 , ..., x_t , ..., x_{T-1}) \in R^{K \times T} (1)$$

où T, est le nombre total de vecteurs de caractéristiques et dépend de la longueur du signal x(m). Les caractéristiques vocales X peuvent être basées sur divers types de représentation, par exemple spectrale (Chen et al., 2014 ; Sainath et Parada, 2015 ; Wang et al., 2019a), cepstrale (Bai et al., 2019 ; Fernández et al., 2007) ou temporelle (Ibrahim et al., 2019).

Le modèle acoustique DNN reçoit X en entrée et produit une séquence de probabilités a posteriori sur les différentes classes de mots-clés et de mots non-clés. Le modèle acoustique utilise séquentiellement les segments $X_{\{i\}}$ de X jusqu'à ce que l'ensemble de la séquence de caractéristiques X soit traité. Les segments consécutifs se chevauchent généralement, mais de nombreux travaux envisagent des modèles acoustiques classant des segments qui ne se chevauchent pas et qui sont suffisamment longs (par exemple, une seconde) pour couvrir un mot-clé entier (Bai et al., 2019 ; Tang et Lin, 2018 ; Zeng et Xiao, 2019 ; Chen et al., 2019 ; Choi et al., 2019 ; Xu et Zhang, 2020 ; Li et al., 2020 ; Yılmaz et al., 2020). En outre, la détection de l'activité vocale est parfois utilisée pour réduire la consommation d'énergie en n'entrant dans le modèle acoustique que les segments $X_{\{i\}}$ dans lesquels la parole est présente (Chen et al., 2015 ; Chen et al., 2014 ; Lugosch et Myer, 2018 ;Tan et al., 2020 ;Yuan et al., 2019 ; Yang et al., 2020).

Le modèle acoustique DNN a N nœuds de sortie signifiant N classes différentes. Normalement, les nœuds de sortie représentent soit des mots (Chai et al., 2019 ; Bai et al., 2019 ; Chen et al., 2014 ; Sainath et Parada, 2015 ; Tang et Lin, 2018 ;Pedroni et al., 2018 ;Sørensen et al., 2020 ; Zeng et Xiao, 2019 ; Chen et al., 2019 ; Choi et al., 2019 ; Xu et Zhang, 2020 ; Li et al., 2020 ; Yılmaz et al., 2020 ; Yang et al., 2020 ;Myer et Tomar, 2018 ;Higuchi et al., 2020) soit des unités de sous-mots telles que des phonèmes indépendants du contexte (Alvarez and Park, 2019 ; He et al., 2017 ; Xuan et al., 2019 ; Sharma et al., 2020). Pour chaque segment d'entrée $X_{\{i\}}$, le modèle acoustique fournit la probabilité à postériori de la n-ième classe .

La plupart des recherches menées sur la détection des mots clés parlés par

l’apprentissage profond se sont concentrées sur la partie clé du système, à savoir la conception de modèles acoustiques de plus en plus précis et de moins en moins complexes sur le plan calculatoire (Rybakov et al., 2020 ; Zhang et al., 2018). De plus, la détection de mots-clés n'est pas une tâche statique mais une tâche dynamique dans laquelle le système de détection doit écouter en permanence le signal d'entrée x(m) pour obtenir la séquence des probabilités à postériori afin de détecter les mots clés en temps réel.

3.2. Extraction des caractéristiques acoustiques

3.2.1. Caractéristiques liées à l'échelle de Mel

Les caractéristiques acoustiques basées sur les bancs de filtres Mel, tels que les coefficients spectraux log-Mel et les coefficients cepstraux de fréquence Mel (MFCC), ont été largement utilisées dans les domaines de la reconnaisssance automatique de la parole et de la détectiondes mots-clés (KWS). Dans les KWS basés sur l’apprentissage profond, les deux types de caractéristiques de la parole sont généralement normalisés pour avoir une moyenne nulle et un écart type unitaire avant d'être introduits dans le modèle acoustique, ce qui permet de stabiliser et d'accélérer l'apprentissage et d'améliorer la généralisation du modèle (LeCun et al., 2012). D’autres caractéristiques tels les MFCC avec contexte temporel et, parfois, leurs dérivés de premier et de second ordre sont utilisés (Bai et al., 2019 ; Tang et Lin, 2018 ; Fernández et al., 2007 ; Xu et Zhang, 2020 ; Li et al., 2020 ; Yılmaz et al., 2020 ;Wöllmer et al., 2013 ; Fuchs et Keshet, 2017 ; Tang et al., 2018 ;Muhsinzoda et al., 2019 ;Pattanayak et al., 2019 ; Chen et al., 2020 ; Berg et al., 2021 ; Wang et al., 2021). Les MFCC sont obtenus par l'application de la transformée en cosinus discrète au spectrogramme log-Mel. Cette transformée produit des caractéristiques approximativement décorrélées, qui conviennent bien à l'apprentissage profond. Toutefois, les modèles d'apprentissage profond sont capables d'exploiter les corrélations spectro-temporelles, ce qui permet d'utiliser le spectrogramme log-Mel au lieu des MFCC et d'obtenir des performances équivalentes ou supérieures des systèmes de ASR et KWS(Watanabe et al., 2017). Par conséquent, un grand nombre de travaux sur les KWS basé sur l’apprentissage profond prennent en compte les caractéristiques log-Mel ou MFCC avec un contexte temporel, comme dans Wang et Long (2018), Shan et al. (2018), Prabhavalkar et al. (2015), Chen et al. (2014), Wang et al. (2019b), Sainath et Parada (2015), Sun et al. (2016), Alvarez R. and H.-J. Park (2019), Sørensen et al. (2020), Wang

et al. (2019a), Zeng et Xiao (2019), Lugosch et Myer (2018), Myer et Tomar (2018), He et al. (2017), Sharma et al. (2020) , Lee et al. (2019), Park et al. (2020), Wu et al. (2020) ,Coucke et al. (2019), Arik et al. (2017), Huang et al. (2019) ,Mazzawi et al. (2019), Yu et al. (2020), Ji et al. (2020), Yan et al. (2020) ,Zhang et Zhang (2020), Kim et al. (2021), Tian et al. (2021). En outre, Gao et al. (2020) proposent d'utiliser la dérivée première du spectrogramme log-Mel pour améliorer la robustesse face aux variations de gain du signal. Le nombre de canaux de bancs de filtres dans les travaux mentionnés ci-dessus va de 20 à 128. Malgré cette large gamme de canaux, les expérimentations ont montré que les performances du système KWS ne sont pas significativement sensibles à la valeur de ce paramètre tant que la résolution de la fréquence Mel n'est pas très faible (López-Espejo et al., 2020a), ce qui pourrait favoriser l'utilisation d'un nombre inférieur de canaux de bancs de filtres afin de limiter la complexité calculatoire.

3.2.2. Caractéristiques basés sur les réseaux de neurones récurrents

Les réseaux de neurones récurrents (RNN) permettent de réduirer des séquences de données de longueur variable en vecteurs de caractéristiques compacts de longueur fixe, également connus sous le nom de "embeddings". Ceci rend les RNN parfaitement convenables aux problèmes de correspondance de modèles (template matching) tels que le (query-by-example KWS), qui consiste en la détection de mots clés en déterminant la similarité entre les vecteurs de caractéristiques, calculés successivement à partir du flux audio d'entrée et les modèles de mots-clés. Dans Chen et al. (2015), Yuan et al. (2019) , Hou et al. (2016), Sacchi et al. (2019), et Huang et al. (2021), les réseaux de neurones LSTM et GRU sont utilisés pour extraire les embeddings de mots. En général, ceux-ci sont comparés, au moyen d'une fonction de distance telle que la similarité cosinus (Singhal, 2001) et en particulier pour les QbE-STD, avec les embeddings de mots clés obtenus au cours d'une phase d’apprentissage.

Dans les QbE-STD basés sur l'extraction de caractéristiques par RNN, la modélisation acoustique est implicitement effectuée par le RNN et elle est particulièrwement utile pour les STD personnalisés à vocabulaire ouvert, qui permettent à un utilisateur de définir ses propres mots clés en enregistrant simplement quelques échantillons de mots clés au cours d'une phase d'inscription. Les QbE-STD basés sur l'extraction de caractéristiques RNN se sont révélés plus efficaces et plus performants que

les approches QbE-STD classiques basées sur le LVCSR Parada et al.(2009) et la distorsion temporelle dynamique (DTW).

3.2.3. Caractéristiques de faible précision

Les recherches récentes Riviello et David (2019) étudièrent deux types de représentations acoustiques de faible précision : le spectrogramme log-Mel quantifié linéairement et la variation de puissance dans le temps, dérivée du spectrogramme log-Mel, représentée par seulement 2 bits. Les résultats expérimentaux montrent que l'utilisation de spectres log-Mel de 8 bits permet d'obtenir la même précision de STD que l'utilisation de MFCC de pleine précision. En outre, la dégradation des performances du STD est insignifiante lorsque l'on exploite des caractéristiques acoustiques d'une précision de 2 bits. Ce fait pourrait indiquer qu'une grande partie des informations spectrales est superflue lorsqu'il s'agit de détecter un ensemble de mots-clés (Riviello et David, 2019 ;López-Espejo et al., 2020a).

3.2.4. Caractéristiques par apprentissage de bancs de filtres

L'apprentissage optimal des bancs de filtres fait partie d'une stratégie de test de bout en bout (end-to-end) et a été étudié pour les KWS profonds (Mittermaier et al., 2020 ;López-Espejo et al. 2020a). Dans ce contexte, les paramètres des bancs de filtres sont réglés de manière à optimiser la génération de mots a posteriori. En particulier, dans Mittermaieret al. (2020), les paramètres du modèle acoustique sont optimisés conjointement avec les fréquences de coupure d'un banc de filtres basé sur les sinc-convolutions (SincConv) Ravanelli et Bengio (2018). De même, dans López-Espejo et al. (2020a) , les auteurs ont utilisé deux approches d'apprentissage de bancs de filtres : l'une consistant en un apprentissage de la matrice des bancs de filtres dans le domaine spectral de puissance et l'autre basée sur l'apprentissage des paramètres de bancs de filtres gammachirp motivée par des considérations psychoacoustiques Irino et Unoki (1999).

3.2.5. Autres caractéristiques acoustiques

Un petit nombre de travaux a exploré l'utilisation d'autres caractéristiques acoustiques ayant un impact calculatoire relativement faible. Par exemple, Ibrahim et al. (2019) ont introduit ce que l'on appelle la similarité temporelle décalée multiframe (MFSTS). Les MFSTS sont des caractéristiques du domaine temporel qui consistent en une représentation

bidimensionnelle du son composée de valeurs d'auto-corrélation à décalage contraint. En dépit de leur simplicité de calcul, pour les applications KWS à faible consommation d'énergie, les caractéristiques telles que les MFCC offrent une bien meilleure précision Ibrahim et al. (2019). Une approche plus intéressante est celle examinée par Shankar et al. (2018), Albert et al. (2019), qui fusionne deux paradigmes KWS différents : DTW et KWS profond. Tout d'abord, une matrice de déformation DTW mesurant la similarité entre un signal audio d'entrée et le modèle de mot-clé est calculée. Du point de vue des KWS profonds, cette matrice peut être considérée comme des caractéristiques acoustiques qui sont transmises à un classifieur binaire d'apprentissage profond (c.-à-d. mot-clé/non-mot-clé) jouant le rôle de "modèle acoustique".

3.3. Modélisation acoustique

3.3.1. Les réseaux de neurones feedforward entièrement connectés

Le premier système de détection de mots clés basé sur l'apprentissage profond (deep KWS) est apparu en 2014 en utilisant une modélisation acoustique basée sur l'architecture neuronale la plus répandue à l'époque: le réseau neuronal feedforward entièrement connecté (FFNN)(Chen et al., 2014). Un simple empilement de trois couches cachées entièrement connectées avec 128 neurones chacune et des activations d'unités linéaires rectifiées (ReLU), suivi d'une couche de sortie softmax, a largement surpassé les performances, avec moins de paramètres.

De nos jours, les modèles acoustiques récents utilisent des réseaux neuronaux convolutifs et récurrents, car ils peuvent fournir de meilleures performances avec moins de paramètres (Shan et al., 2018 ;Sainath et Parada, 2015). Malgré cela, les modèles acoustiques FFNN standard et leurs variantes sont pris en compte dans la littérature récente, soit à des fins de comparaison, soit pour l'étude des aspects des KWS tels que les fonctions de perte d'apprentissage (Shan et al., 2018 ; Hou et al., 2019 ; Liu et al., 2019 ; Yuan et al. 2019) . Les alternatives étroitement liées et moins coûteuses en termes de calcul aux FFNN entièrement connectés sont le filtre à décomposition de valeur unique (SVDF) (Alvarezand Park, 2019 ; Park et al., 2020 ;Nakkiran et al., 2015) et les réseaux de neurones à pointes (Pedroni et al., 2018 ; Yılmaz e al., 2020 ; Mostafa, 2018). Ces alternatives ont permis la réduction de la taille du premier KWS profond sans baisse de

performance.

Les réseaux de neurones à pointes (SNN) sont inspirés du cerveau humain qui, contrairement aux réseaux de neurones artificiels (ANN), traitent les informations de manière événementielle, ce qui allège considérablement la charge de calcul lorsque ces informations sont peu nombreuses, comme dans les KWS (Pedroni et al., 2018 ; Yılmaz et al., 2020 ; Mostafa, 2018).

3.3.2. Les réseaux de neurones convolutifs

Le passage d'un FFNN entièrement connecté à la modélisation acoustique par CNN a été introduite en 2015 par Sainath et Parada (2015). Grâce à l'exploitation des corrélations temps-fréquence locales du son, les CNN sont capables de surpasser, avec moins de paramètres, les FFNN entièrement connectés pour la modélisation acoustique dans les KWS profonds (Sainath et Parada, 2015 ;Rybakov et al., 2020 ; Wu et al., 2020 ; Tang et al., 2018 ; Yu et al., 2020 ; Shankar et al., 2018 ; Huang et al., 2018 ; Menon et al., 2018 ; Liu et al., 2020 ; Mo et al., 2020). L'une des caractéristiques des CNN est que le nombre de multiplications du modèle peut être facilement limité pour répondre aux contraintes de calcul en ajustant différents hyperparamètres comme, par exemple, le pas du filtre et les tailles du noyau et du pooling. En outre, cela peut se faire sans nécessairement perdre en performances.

L'apprentissage résiduel, proposé par He et al. (2016) pour la reconnaissance d'images, est largement considéré comme l'état de l'art de la mise en œuvre de modèles acoustiques pour les KWS profonds. En bref, les modèles d'apprentissage résiduel sont construits en introduisant une série de connexions de raccourci reliant des couches non consécutives ce qui permet de mieux apprendre des modèles CNN profonds. Tang et Lin (2018) ont été les premiers auteurs à explorer l'apprentissage résiduel profond pour les KWS profonds. Ils ont également intégré des convolutions dilatées augmentant le champ réceptif du réseau afin de capturer des segments temps-fréquence plus longs sans augmenter le nombre de paramètres. De cette manière, Tang et Lin ont largement surpassé, avec moins de paramètres, les CNN standard de Sainath et Parada (2015) en termes de performances KWS, établissant ainsi un nouvel état de l'art en matière de KWS en 2018.

Les CNN séparables en profondeur (DS-CNN) sont un bon choix pour mettre en œuvre des modèles acoustiques performants dans les systèmes embarqués Sørensen et al.

(2020), Wang et al. (2019a) et Fernández et al. (2007). En outre, la combinaison de convolutions séparables dans le sens de la profondeur avec l'apprentissage résiduel a été récemment explorée pour la modélisation acoustique profonde des KWS (Xu et Zhang, 2020 ; Li et al., 2020 ; Yang et al., 2020 ; Kim et al., 2021).

Un modèle acoustique basé sur les CNN devrait englober les trois aspects suivants López-Espejo et al. (2022) ;

1. Un mécanisme permettant d'exploiter les longues dépendances temps-fréquence comme, par exemple, l'utilisation de convolutions temporelles ou des circonvolutions dilatées(Choi et al., 2019).
2. Convolutions séparables en profondeur pour réduire considérablement l'empreinte mémoire et les coûts de calcul(Howard et al., 2017).
3. Connexions résiduelles pour former rapidement et efficacement des modèles plus profonds qui améliorent les performances des KWS(He et al., 2016).

3.3.3. Les réseaux de neurones réccurents et à retardement

La parole est une séquence temporelle avec de fortes dépendances temporelles. Par conséquent, l'utilisation des réseaux de neurones récurrents (RNN) pour la modélisation acoustique ainsi que des réseaux de neurones à retardement (TDNN), qui sont formés par un ensemble de couches fonctionnant à différentes échelles temporelles, s'impose naturellement. Par exemple, les réseaux LSTM (pour long short term memory) (Hochreiter et. Schmidhuber, 1997), qui surmontent les problèmes d'explosion et de disparition du gradient rencontrés par les RNN standard, sont utilisés pour la modélisation acoustique des KWS, notamment dans Zhuang et al. (2016), Sun et al. (2016), Kumar et al. (2018), Coucke et al. (2019), Wöllmer et al. (2013), avec des performances nettement supérieures à celles des FFNN (Sun et al., 2016).

Lorsque la latence n'est pas une contrainte forte, des LSTM bidirectionnelles (BiLSTM) peuvent être utilisées à la place pour capturer les dépendances causales et anticausales afin d'améliorer les performances des KWS (Kumar et al., 2018 ; Sundar et al.,2015). Par ailleurs, les GRU (pour gated reccurent unit) bidirectionnels sont étudiés dans Rybakov et al. (2020) pour la modélisation acoustique des KWS. Lorsqu'il n'est pas nécessaire de modéliser des dépendances temporelles très longues, comme c'est le cas dans les KWS, les GRUs peuvent être préférées aux LSTMs car les premières demandent moins

de mémoire et sont plus rapides à entrainer tout en ayant des performances similaires ou même meilleures (Arik et al., 2017). Myeret Tomar (2018) étudièrent un TDNN en deux étapes composé d'un modèle acoustique LVCSR suivi d'un classifieur de mots clés.

Myeret Tomar (2018) étudièrent également l'intégration du saut de trame et de la mise en cache pour réduire les calculs, ce qui permet de surpasser la modélisation acoustique CNN classique de Sainath et Parada (2015) tout en réduisant de moitié le nombre de multiplications.

Les CNN peuvent avoir des difficultés à modéliser les dépendances temporelles à long terme. Pour surmonter ce problème, ils peuvent être combinés avec des RNN pour construire ce que l'on appelle les CRNN. Ainsi, on peut dire que les CRNN offrent le meilleur de deux mondes : tout d'abord, les couches convolutives modélisent les corrélations spectro-temporelles locales de la parole et, ensuite, les couches récurrentes font de même en modélisant les dépendances temporelles à long terme dans le signal de la parole.

Certains travaux explorent l'utilisation des CRNN pour la modélisation acoustique dans les KWS parlés profonds en utilisant des LSTM unidirectionnelles ou bidirectionnelles ou des GRU (Rybakov et al., 2020 ; Zeng et Xiao, 2019 ; Kumar et al., 2018 ; Arik et al., 2017 ; Liu et Sun, 2019 ; Albert et al., 2019).

3.4. Classification temporelle connexionniste

Les modèles acoustiques RNN sont généralement entrainés pour produire des probabilités a posteriori au niveau fenêtre. Au moment de l'apprentissage, en cas d'utilisation, par exemple, de la perte d'entropie croisée, des données annotées au niveau fenêtre sont nécessaires, ce qui peut être difficile à obtenir. Dans le contexte de la modélisation acoustique des RNN, la classification temporelle connexionniste (CTC) (Graves et al., 2006) permet au modèle de localiser et d'aligner de manière non supervisée les étiquettes des unités phonétiques au moment de l'apprentissage (Zhuang et al., 2016). En d'autres termes, les alignements au niveau fenêtre des séquences d'étiquettes cibles ne sont pas nécessaires pour l'apprentissage.

La toute première tentative d'application de la CTC aux KWS a été réalisée par Fernández et al.(2007) en utilisant un BiLSTM pour la modélisation acoustique. Au moment de l'apprentissage, ce système a simplement besoin, avec les signaux audio

d'apprentissage, de la liste des mots d'apprentissage dans l'ordre d'apparition. Après cette première tentative, plusieurs travaux ont exploré des variantes de cette approche en utilisant différentes architectures de RNN comme les LSTM (Zhuang et al., 2016 ; He et al., 2017 ; Xuan et al., 2019 ; Bai et al., 2016), les BiLSTM (Wöllmer et al., 2013 ; Yan et al., 2020) et les GRU Xuan et al., 2019 ; Ceolini et al., 2019), ainsi qu'en considérant différentes unités phonétiques comme les phonèmes (He et al., 2017 ; Wöllmer et al., 2013) et les syllabes en mandarin (Wang et Long, 2018 ; Bai et al., 2016).

3.5. Modèles de séquence à séquence

La CTC suppose l'indépendance conditionnelle des étiquettes, c'est-à-dire que les sorties antérieures du modèle n'influencent pas les prédictions actuelles. Par conséquent, dans le contexte de KWS et de l'ASR en général, la CTC peut avoir besoin d'un modèle linguistique externe pour être performant. Par conséquent, une approche plus pratique pour la modélisation acoustique des KWS pourrait être l'utilisation de modèles séquence à séquence (Seq2Seq), proposés pour la première fois dans Sutskeveret al. (2014) pour la traduction linguistique.

Les modèles Seq2Seq se composent d'un encodeur RNN qui résume la séquence d'entrée de longueur variable en un vecteur de dimension fixe, suivi d'un décodeur RNN qui génère une séquence de sortie de longueur variable conditionnée à la fois par la sortie de l'encodeur et par les prédictions passées du décodeur. Outre les tâches connexes telles que QbE-STD(Chung et al., 2016), les modèles Seq2Seq tels qu'un RNN-Transducteur (RNN-T) ont également été étudiés pour les KWS parlés profonds (He et al., 2017 ; Sharma et al., 2020 ; Tian et al., 2021 ; Liu et al., 2021). Le RNN-T, intégrant à la fois des modèles acoustiques et linguistiques (et prédisant les phonèmes), est capable de surpasser un système KWS CTC même lorsque ce dernier exploite un modèle linguistique N-gramme de phonème externe (He et al., 2017).

3.6. Le mécanisme de l'attention

Dans les modèles Seq2Seq, le codeur doit condenser toutes les informations nécessaires dans un vecteur de dimension fixe, quelle que soit la longueur (variable) de la séquence d'entrée, ce qui peut s'avérer difficile. Le mécanisme d'attention (Vaswani et al., 2017), semblable à l'attention d'écoute humaine, pourrait aider dans ce contexte en se concentrant

sur les sections du discours qui sont plus susceptibles de contenir un mot-clé (Shan et al., 2018). L'intégration d'un mécanisme d'attention (y compris une variante appelée attention multi-têtes Vaswani et al., 2017) dans les modèles acoustiques Seq2Seq afin de se concentrer sur le(s) mot(s) clé(s) d'intérêt a été réalisée avec succès par un certain nombre de travaux (Wang et al., 2019b ; Rybakov et al., 2020 ; He et al., 2017 ; Lee et al., 2019 ; de Andrade et al. , 2018 ; Liu et al., 2021 ; Zhao et Zhang, 2020). Ces travaux ont montré que l'intégration de l'attention permet aux KWS des gains de performance par rapport aux modèles Seq2Seq homologues sans attention.

Il est à noter que l'attention a également été étudiée en conjonction avec les TDNN pour les KWS (Chai et al., 2019 ; Bai et al., 2019). En particulier, dans Bai et al. (2019), grâce à l'exploitation de l'auto-attention des poids partagés, Bai et al. reproduisent les performances du modèle d'apprentissage résiduel profond res15 de Tang et Lin (2018) en utilisant beaucoup moins de paramètres.

3.7. Apprentissage du modèle acoustique

Une fois que l'architecture du modèle acoustique a été conçue ou "recherchée" de manière optimale, l'estimation de ses paramètres se fait de manière discriminante selon un critère d'optimisation , défini par une fonction de perte, au moyen de la rétropropagation et en utilisant des données audio étiquetées/annotées (Mazzawi et al., 2019 ; Zhang et al., 2021).

3.7.1. Fonctions de perte

En dehors de la CTC, la perte d'entropie croisée est la fonction de perte la plus populaire pour l'apprentissage de modèles acoustiques profonds de KWS parlés (Bridle, 1990 ; Goodfellow et al., 2016). Par exemple, la perte d'entropie croisée est considérée par Chai et al. (2019), Bai et al. (2019), Chen et al. (2014), Sun et al. (2016), Tang et Lin (2018), Alvarez R. and H.-J. Park (2019), Rybakov et al. (2020), Liu et al. (2019), Sørensen et al. (2020), Kumar et al. (2018), Arik et al. (2017), Tucker et al. (2016), Menon et al. (2018) , Liu et al. (2020). Lorsque le modèle acoustique est destiné à produire des probabilités à posteriori au niveau sous-mots, les étiquettes d'apprentissage sont généralement générées par un alignement forcé à l'aide d'un système LVCSR, ce qui conditionne les performances ultérieures du système KWS (Chen et al., 2014 ; Alvarez et Park, 2019 ; Liu et al., 2019).

La perte max-pooling (Scherer et al., 2010), qui est une alternative à la perte

d'entropie croisée, a également été étudiée à des fins de KWS (Sun et al., 2016 ; Hou et al., 2020 ; Park et al., 2020). Dans le contexte des KWS, l'objectif de la perte max-pooling est d'apprendre au modèle acoustique à ne se déclencher qu'au moment où la confiance est la plus élevée, vers la fin du mot-clé. La perte max-pooling s'est avérée plus performante que la perte d'entropie croisée en termes de performances des KWS, en particulier lorsque le modèle acoustique est initialisé par la perte d'entropie croisée pour l'apprentissage (Sun et al., 2016).

Des variantes de perte max-pooling faiblement contrainte et lissée sont proposées respectivement dans Hou et al. (2020) et Park et al. (2020), ce qui permet de réduire la dépendance à l'égard de la précision de l'alignement forcé LVCSR.

3.7.2. Paradigmes d'optimisation

Dans les KWS profonds, les optimiseurs les plus fréquemment utilisés sont la descente stochastique du gradient (SGD) (Kiefer et Wolfowitz, 1952) comme dans Tang et Lin (2018), Alvarez et Park (2019), Chen et al. (2019),Choi et al. (2019),Xu et Zhang (2020), Yılmaz et al. (2020), Kumar et al. (2018), Wang et al. (2021), Kim et al. (2021), Tucker et al. (2016), Sundar et al. (2015), Liu et al. (2021), Zhang et al. (2021), et Kingma et Ba (2015).

Il est également courant de mettre en œuvre un mécanisme qui réduit le taux d'apprentissage au fil des époques (Shan et al., 2018 ; Chai et al., 2019 ; Bai et al., 2019 ; Chen et al., 2014 ; Sun et al., 2016 ; Sørensen et al., 2020 ; Zeng et Xiao, 2019 ; Chen et al., 2019 ; Xu et Zhang, 2020 ; Yılmaz et al., 2020 ; Lee et al., 2019 ; Mittermaier et al., 2020 ; Kumar et al., 2018 ; Tucker et al., 2016 ; An et al., 2019).

En outre, de nombreux travaux sur les KWS profonds, par exemple Shan et al. (2018), Chen et al. (2019), Choi et al. (2019) , Xu et Zhang (2020), Berg et al. (2021), Kim et al. (2021), Liu et al. (2021), déploient une forme de régularisation des paramètres comme la décroissance et l'abandon des poids.

Alors que l'initialisation aléatoire des paramètres du modèle acoustique est l'approche normale, l'initialisation basée sur l'apprentissage par transfert à partir des modèles acoustiques LVCSR s'est avérée conduire à de meilleurs modèles KWS en atténuant, par exemple, le sur-ajustement (Chen et al., 2014 ; Myeret Tomar, 2018 ; Tian et al., 2021).

3.7.3. Traitement des probabilités à posteriori

Afin de parvenir à une décision finale sur la présence ou non d'un mot-clé dans un flux audio, la séquence des probabilités produite par le modèle acoustique doit être traitée. Il existe deux modes principaux de traitement des postériorités : le mode sans flux (statique) et le mode avec flux (dynamique) (López-Espejo et al., 2022).

A. Mode sans flux (non-streaming)

Le mode sans flux fait référence à la classification multi-classe standard de segments d'entrée indépendants comprenant un seul mot chacun (c'est-à-dire la classification de mots isolés).

Pour couvrir la durée d'un mot entier, les segments d'entrée doivent être suffisamment longs, par exemple environ 1 seconde (Warden, 2017 ; Warden, 2018). Dans ce mode, étant donné un segment d'entrée X{i} , celui-ci est assigné à la classe ayant la probabilité a posteriori la plus élevée. Cette approche est préférable à la sélection de classes dont les probabilités a posteriori sont supérieurs à un seuil de sensibilité à fixer. Les expérimentations de López-Espejo et al. (2020a),López-Espejo et al. (2019), López-Espejo et al. (2020b), López-Espejo et al. (2021) montrent que les systèmes KWS profonds sans flux ont tendance à produire des distributions a posteriori très élevées. Le mode sans flux a été utilisé dans les systèmes KWS profonds pour la classification des mots isolés de Bai et al. (2019), Tang et Lin (2018), Rybakov et al. (2020), Zeng et Xiao (2019), Chen et al. (2019),Choi et al. (2019) , Xu et Zhang (2020), Li et al. (2020, Myeret Tomar (2018), Riviello et David (2019), López-Espejo et al. (2020a) , Chen et al. (2020), Zhang et Zhang (2020), Liu et Sun (2019), Mo et al. (2020), López-Espejo et al. (2019), López-Espejo et al. (2020b), López-Espejo et al. (2021) du fait que le cadre expérimental est plus simple.

B. Mode avec flux (streaming)

Le mode avec flux fait référence au traitement continu d'un flux audio d'entrée dans lequel les mots-clés ne sont pas isolés/segmentés. Par conséquent, dans ce mode, un segment donné peut ou non contenir un (ou une partie) mot-clé. Dans ce cas, le modèle acoustique produit une séquence temporelle de probabilités a posteriori avec de fortes corrélations locales. Pour cette raison, la séquence de probabilités a posteriori brutes, qui est

intrinsèquement bruyante, est généralement lissée dans le temps , par exemple par une moyenne variable, sur la base d'une classe avant la suite du traitement (Prabhavalkar et al., 2015 ; Chen et al., 2014 ; Sun et al., 2016 ; Liu et al., 2019 ; Sørensen et al., 2020 ; Wang et al., 2019a ; Yuan et al., 2019 ; Myer et Tomar, 2018 ; Wu et al., 2020 ; Kumar et al., 2018 ; Tan et al., 2019).

Ensuite, les probabilités a posteriori lissées des mots sont souvent utilisées directement pour déterminer la présence ou non d'un mot-clé, soit en les comparant à un seuil de sensibilité (Sun et al., 2016 ; Sørensen et al.,2020 ; Myeret Tomar, 2018), soit en choisissant, dans une fenêtre glissante temporelle, la classe ayant la probabilité a posteriori la plus élevée (Kumar et al., 2018). Il convient de noter que les segments d'entrée consécutifs peuvent couvrir des fragments de la même réalisation de mot clé. De fausses alarmes peuvent se produire en raison de la reconnaissance de la même réalisation du mot clé plusieurs fois à partir de la séquence de probabilités a posteriori lissée. Pour éviter ce problème, un mécanisme simple consiste à forcer le système KWS à ne pas se déclencher pendant une courte période juste après qu'un mot-clé ait été repéré (Sun et al., 2016 ; Sørensen et al., 2020).

Dans le cas où les mots clés sont composés de plusieurs mots (détection de termes parlés) ou chacune des N classes représente un sous-mot, le système proposé dans Chen et al. (2014) présente une méthode qui consiste à traiter les probabilités a posteriori lissées afin de produire une décision sur la présence ou non du mot-clé.

Un mot clé est détecté chaque fois que le score de confiance dépasse un seuil de sensibilité à définir. Cette approche a été utilisée dans les KWS profonds comme Wang et al. (2019a), Yuan et al. (2019) , Tan et al. (2019). Dans Prabhavalkar et al. (2015), la contrainte que les sous-unités du mot clé se déclenchent dans le bon ordre d'occurrence au sein du mot clé est ajoutée, ce qui contribue à réduire les fausses alertes. Cette version améliorée de la méthode de traitement a posteriori est prise en compte dans Liu et al. (2019), Wu et al. (2020).

Lorsque chacune des N classes d'un système KWS profond représente une unité de sous-mot telle qu'une syllabe ou un phonème indépendant du contexte, un treillis consultable peut être construit à partir de la séquence temporelle des probabilités a postériori. Cela se fait généralement dans le contexte de la CTC (Zhuang et al., 2016 ; Wang et Long, 2018).

L'objectif est alors de trouver, à partir du treillis, la séquence d'unités de sous-mots la plus similaire à celle du mot-clé cible. Si le score résultant de la recherche sur le treillis est supérieur à un seuil prédéfini, un mot-clé est repéré.

Chapitre 4 : Une Approche acoustique pour la détection de mots parlés

4.1. Introduction

La détection de mots parlés (STD) pour interroger un flux de parole est une tâche imminemment non supervisée, et comme nous l'avons présentement introduit, elle est souvent abordée en utilisant des métriques de distance entre la requête parlée et l'archive parole à prospecter. La recherche d'une partie du flux qui se rapproche le plus de la requête parlée est souvent réalisée par un algorithme relevant de la programmation dynamique, le plus souvent le DTW ou l'une de ses variantes.

Dans cette première contribution, nous proposons de détecter la présence d'un mot parlé dans un flux de parole par une méthode non supervisée qui peut se généraliser à la détection de mots parlés dans un système de commande vocale ou dans un contexte multilingue.

Il s'agit d'une proposition basée sur le DTW, avec une segmentation non pas uniforme mais qui se réfère à la segmentation du signal sur la base l'énergie, ainsi, nous nous assurons que nous obtenons des segments de parole qui se rapprochent le plus des mots ou de parties de mots significatives (Benati et Bahi, 2016). Ces segments identifiés sont comparés au mot recherché pour déterminer s'ils s'y apparentent ou non.

Pour montrer l'aspect indépendant de la langue, la proposition est testée sur les enregistrements de récits en Arabe et en Anglais issus de la base de parole de l'IPA (International Phonetic Association) accessible à partir de la page : https://www.internationalphoneticassociation.org/content/ipa-handbook-downloads.

4.2. Techniques de mise en correspondance

4.2.1. La déformation temporelle dynamique

La déformation temporelle dynamique (DTW) est utilisée pour trouver la similarité entre deux modèles en déformant l'axe temporel d'un modèle pour qu'il corresponde à l'autre.

La DTW permet de trouver une région dans l'énoncé de test qui est très similaire à l'échantillon de requête. La déformation temporelle peut gérer les variabilités du débit de parole dans deux modèles de parole. La mise en correspondance des modèles peut être effectuée soit en utilisant des caractéristiques basées sur le spectre (comme MFCC, PLP) soit en utilisant d'autres caractéristiques symboliques comme les caractéristiques postérieures. De nombreuses variations sont apportées à l'algorithme DTW de base pour le rendre plus puissant pour les applications de recherche audio.

4.2.1.1. DTW de base

Considérons deux modèles de requête : Q = (q1 , q2 , . . . , qi , . . . , qn) avec n trames de caractéristiques et test : T = (t1 , t2 , . . . , tj , . . . , tm), modèles avec m trames de caractéristiques. DTW trouve une fonction de déformation n = ω(m), afin de mapper l'axe temporel m de T avec n de Q. Ceci conduit à une matrice de distance (m × n).

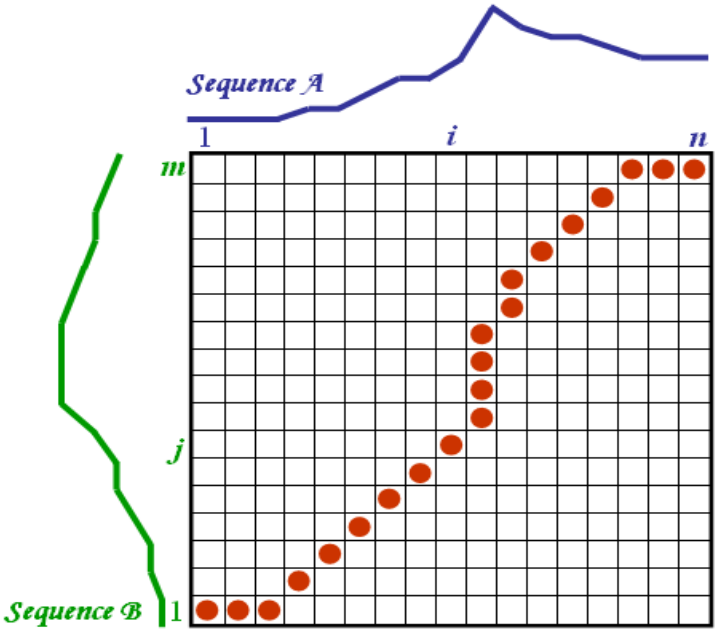


Figure 4.1 : Exemple illustratif de l'alignement temporel antre deux séquences

Dans la matrice construite, chaque case représente le coût d'aligner deux points spécifiques des deux séquences.

Un chemin optimal est trouvé dans cette matrice, du coin inférieur gauche au coin supérieur droit. Ce chemin minimise le coût total de l'alignement. Ce chemin optimal

représente la déformation temporelle nécessaire pour aligner au mieux les deux séquences. D étant la mesure de distance minimale correspondant au meilleur chemin ω(m) à travers la grille de (m × n) points.

A. Contraintes globales

La contrainte globale permet de réduire considérablement la complexité de calcul en limitant la fenêtre de déformation. Cela limite l'étirement excessif de la requête ou du test ou des deux afin d'obtenir un alignement.

B. Contraintes locales

Les contraintes locales telles que le type d'arc utilisé dans le chemin de déformation et le poids associé à chaque branche de l'arc sont prises en compte. Le type d'arc définit le flux du chemin de déformation de l'image de départ à l'image de fin.

4.2.1.2. Les varianbtes du DTW

Les problèmes avec le DTW de base sont son temps de calcul ingérable et ses besoins en mémoire. De plus, il compare deux séquences de longueur presque égale, mais en pratique les deux séquences peuvent avoir des longueurs différentes. Par conséquent, des variantes du DTW sont suggérées par les chercheurs.

A. DTW à point final contraint

Le chemin de déformation doit inclure à la fois les points de départ et de fin des séquences. Par conséquent, aucune liberté n'est autorisée dans la correspondance entre la première et la dernière trame des modèles. C'est ce qu'on appelle le DTW à point final contraint (CE-DTW). Il est utile pour les systèmes dont les performances dépendent de l'exactitude de la détection du point final.

B. DTW à point final non contraint

Le DTW à point final non contraint (UE-DTW) est une variante du DTW qui permet des relaxations de contraintes locales jusqu'à « x » trames, mais uniquement aux emplacements de début et de fin. En modifiant la valeur de la plage de relaxation « x », différentes variations du DTW UE peuvent être envisagées en fonction des applications. Toutes ces

méthodes permettent à la fonction de déformation de démarrer à partir d'un point autre que (1, 1) et de se terminer à un point autre que (m, n).

C. DTW modifié

Il permet de trouver un chemin bien adapté entre la requête et l'énoncé de test. Deux contraintes sont incorporées ici. La première contrainte interdit les extensions de chemin multitrame simultanées dans la requête et l'énoncé de test. Si un chemin DTW progresse jusqu'à l'index i dans la requête et j dans la séquence de test, il peut alors être étendu à i + n et j + m, respectivement, avec une contrainte selon laquelle n = 1 ou m = 1.

La deuxième contrainte consiste à favoriser les extensions de chemin avec des durées similaires en mettant à l'échelle le score de similarité. Le score de distance de l'extension de chemin est normalisé par le nombre de trames m prises par le côté test de l'extension. La mise à l'échelle peut être effectuée à l'aide d'un facteur de pente d'alignement qui peut être contrôlé de manière exponentielle par la variable de contrainte de durée. Le DTW trouve le chemin de score minimal à travers la matrice de similarité.

D. DTW segmentaire (S-DTW)

Considérons Q = (q1 , q2 , . . . , qi , . . . , qn) et T = (t1 , t2 , . . . , tj , . . . , tm) comme la séquence de caractéristiques des séquences de requête et de test, respectivement. La longueur de la caractéristique de test est généralement très supérieure à celle de la requête ($m \geq n$). L'objectif est de trouver une sous-séquence X(a' ; b') telle que $1 \leq a' \leq b' \leq m$. Le DTW segmentaire (S-DTW) tente de trouver des requêtes ou des séquences de type requête à partir des données de test. Le S-DTW définit deux contraintes, à savoir la contrainte globale et la contrainte locale.

C. DTW segmentaire modifié

Diverses formes modifiées de DTW segmentaire ont été introduites pour réduire l'espace de recherche et la complexité temporelle. Le DTW segmentaire normalisé localement (SLN-DTW) est une variante du S-DTW, où les vecteurs de caractéristiques acoustiquement similaires sont regroupés en un seul vecteur et sont considérés comme un segment. Le S-DTW est en outre modifié à l'aide du clustering agglomératif. Le DTW à sous-séquence efficace en mémoire (MES-DTW) est une autre variante du S-DTW. Au

lieu de faire du S-DTW, la complexité de calcul peut être réduite en faisant la moyenne des caractéristiques.

D. DTW non segmentaire

La segmentation de l'audio parlé suivie du DTW est plus coûteuse en termes de calcul. Par conséquent, une variante du DTW appelée DTW non segmentaire (NS-DTW) a été introduite.

Seule la contrainte locale est utilisée ici. Ensuite, une matrice de similarité S de taille m × n est calculée. Le terme de requête est susceptible de commencer à partir de n'importe quel point dans les données de test. Une approche de correspondance partielle est proposée pour récupérer les requêtes qui n'apparaissent pas exactement dans les données de recherche.

4.2.1.3. Distance d'édition minimale

La « distance d'édition » (en Anglais String distance) est une mesure permettant de mesurer la similarité entre deux chaînes de texte. La distance d'édition minimale (MED) et ses variantes sont les algorithmes les plus populaires pour trouver les distances de chaîne.

La MED et ses variantes sont utilisées dans la recherche audio où les fichiers audio sont d'abord convertis en messages texte correspondants à l'aide de systèmes ASR. La MED entre deux chaînes (séquences de requête et de test) est le nombre minimal d'opérations d'édition (insertion, substitution et suppression) nécessaires pour transformer une chaîne en une autre.

A. MED conventionnel

La distance minimale d'édition (MED) est définie comme le coût minimum de conversion d'une chaîne en une autre à l'aide des trois opérations de base [insertion (I), substitution (S) et suppression (D)].

La MED est calculée par programmation dynamique. Une matrice de distance est calculée avec chaque symbole d'une séquence disposé le long d'une ligne et celui de l'autre séquence le long d'une colonne. Cette matrice est appelée matrice de distance d'édition. Chaque cellule de la matrice de distance d'édition peut être calculée comme une simple fonction des cellules environnantes. En partant du début de la matrice, il est possible de remplir chaque cellule de la matrice. La valeur de chaque cellule est calculée en prenant le

minimum des trois chemins possibles (insertion, substitution et suppression).

B. MED modifié

Dans la mesure MED conventionnelle, les pénalités de substitution sont calculées selon des règles heuristiques basées sur les classes. Dans la mesure MED modifiée, les pénalités de substitution sont dérivées de la matrice de confusion phonétique du dispositif de reconnaissance, car la confusion entre différents phonèmes peut être facilement apprise à partir d'une matrice de confusion. Il a été rapporté dans Audhkhasi et Verma (2007) que la recherche par mot-clé devient plus précise si nous utilisons la mesure MED modifiée au lieu de la mesure MED conventionnelle.

4.3. Approche Proposée

Pour aborder le problème de la détection de mots parlés par une approche non supervisée, nous proposons un système qui inclut les étapes suivantes : La segmentation du flux dans lequel nous allons rechercher le mot cible, la segmentation se fait sur la base de l'énergie du signal. Ensuite, le mot cible ainsi que les segments obtenus sont représentés par les MFCCs correspondants. Après, on compare la représentation du mot cible avec celles de tous les segments obtenus, et enfin sur la base du score de similitude obtenu, décider que le segment contient ou non le mot ciblé. Dans le cas où le mot clé est détecté, le système est capable de localiser le mot recherché.

La figure suivante nous donne une vue générale des différentes étapes induites par notre proposition. Cette approche, nous permet de réaliser la détection selon une approche non supervisé, car le système n'a aucune connaissance à priori du mot cible à détecter.

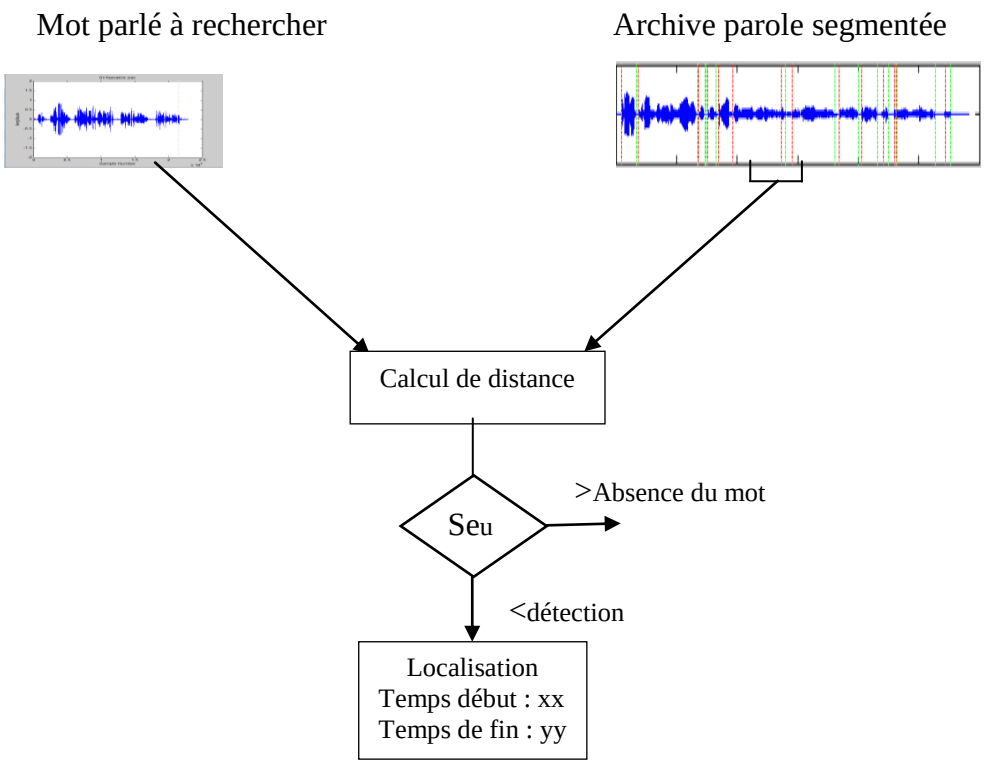


Figure 4.2 :Architecture du système de détection mots clés proposé

4.3.1. Segmentation

D'abord, nous considérons l'archive audio cible que nous devons prospecter à la recherche du terme parlé. Le signal associé est segmenté en se basant sur son énergie. Nous utilisons en guise de mesure de l'énergie, l'énergie à court terme (STE pour Short Term Energy) qui pour les signaux de parole reflète la variation d'amplitude. Dans un signal de parole typique, nous pouvons voir que certaines de ses propriétés changent considérablement avec le temps, et que l'énergie associée au signal de la parole varie au cours du temps. Ainsi, tout traitement de la parole va s'intéresser à la manière dont cette énergie varie.

A titre d'exemple, le signal de la parole consiste en des zones voisées, non voisées ou du silence. Les zones voisées ont une énergie bien plus grande que celles des zones non voisées, tandis que le silence a une énergie presque nulle. Ainsi, l'énergie à court terme est utilisée pour la classification du signal de la parole en zone voisée, non voisée et en silence, cette classification permet de distinguer les voyelles des consonnes ; les voyelles ayant un voisement plus élevé.

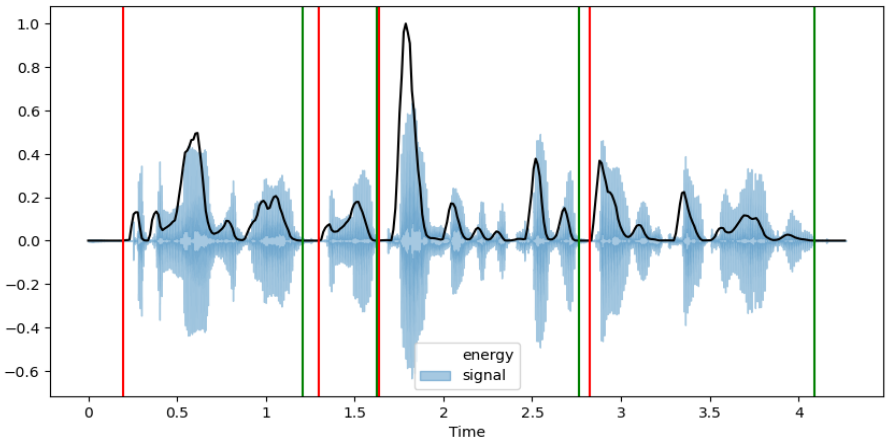
Dans notre cas, l'énergie, nous permet de récupérer les zones du signal qui sont susceptibles de contenir les mots, et qui seront comparées au mot cible.L'énergie à court terme du signal de la parole est calculée dans le domaine du temps où le signal est fenêtré et en calculant la moyenne sur les échantillons élevé au carré.

$$E_n = \frac{1}{N}\sum_{m=1}^{N}[x(m)w(n-m)]^2 \qquad (4.1)$$

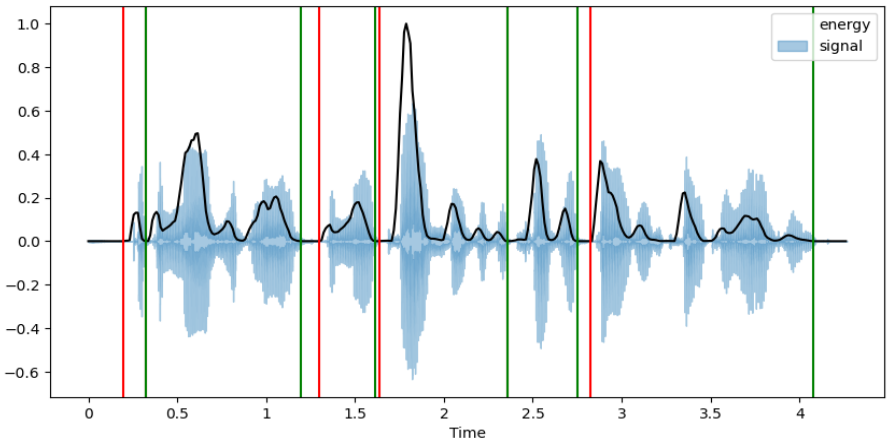
La segmentation sur la base de l'énergie suppose une comparaison avec un seuil en deçà duquel le contenu du signal est supposé ne pas être de la parole. De la valeur de ce seuil dépendra la pertinence de notre segmentation. Dans notre cas, et comme les ensregistrement à prospecter proviennent d'environnement différents où l'intensité du signal varie, nous optons pour un calcul de seuil dynamique qui est fonction de l'énergie moyenne de l'enregistrement en question.

A titre d'exemple, la figure 4.2. est une illustration la segmentation du fichier wav « nothwind5.wav » accessible à la page : https://www.mq.edu.au/about/about-the-university/our-faculties/medicine-and-health-sciences/departments-and-centres/department-of-linguistics/our-research/phonetics-and-phonology/speech/australian-english-pronunciation-and-transcription/the-north-wind-and-the-sun.

Le texte associé à ce signal est:« *but the more he blew the more closely did the traveller fold his cloak around him* ». Pour cette illustration, nous avons adossé le seuil à la valeur de l'énergie minimale dans le signal. On obtient les segmentation suivantes en fonction de celle-ci. Dans le premier cas, la valeur du seuil est de 500 fois celle de l'énergie minimale, et dans le second cette valeur est de 1500 fois celle de l'énergie minimale.



(a)

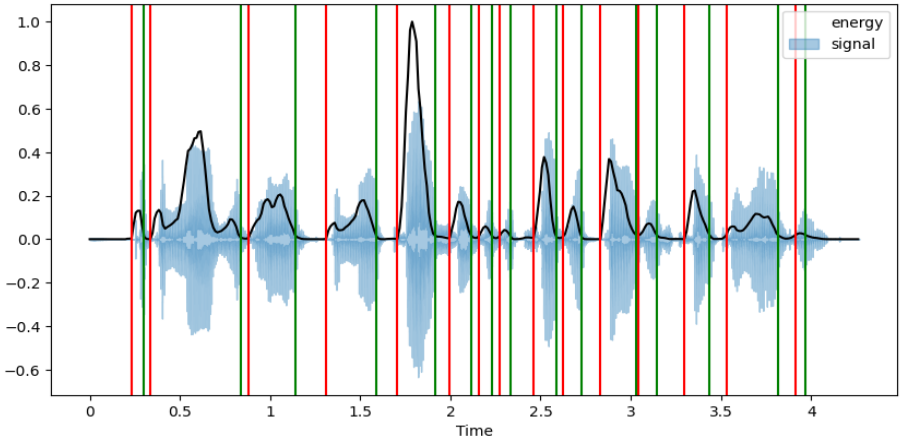


(b)

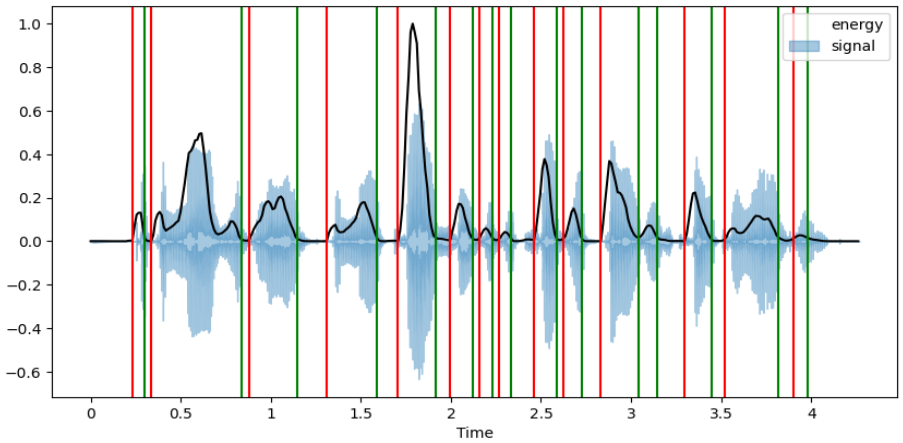
Figure 4.3 : Exemple de segmentation adossée à l'énergie minimale du signal

Dans cette figure, les lignes rouges représentent le début d'un terme et les vertes la fin de ce terme. Dans le premier cas, on obient quatre segments de parole, tandis que dans le second, on a six segments. En fait, dans les deux cas la valeur de l'énergie minimale du signal est tellement proche de zéros que la considérer comme référence conduit à une segmentation inappropriée.

Ainsi, dans notre cas, nous avons considéré à la fois la valeur minimale mais aussi la valeur moyenne de l'énérgie dans le signal. La figure 4.3. illustre la segmentation du même signal en considérant comme seuil le tiers puis le quart de la différence entre la valeur moyenne et la valeur minimale.



(a)



(b)

Figure 4.4. Exemples de segmentation adossée à l'énergie moyenne et à l'énergie minimale

4.3.2. La représentation acoustique

Pour extraire une représentaton pertinente des différents segments de parole considérés, nous procédons à la représentation de ces segments en utilisant les coefficients MFCCs. La figure 4.4. illustre la représentation du signal « narrative1.wav » par les coefficients MFCCs.

Une fois segmenté, les différents segments obtenus sont comparé un à un au terme cible.

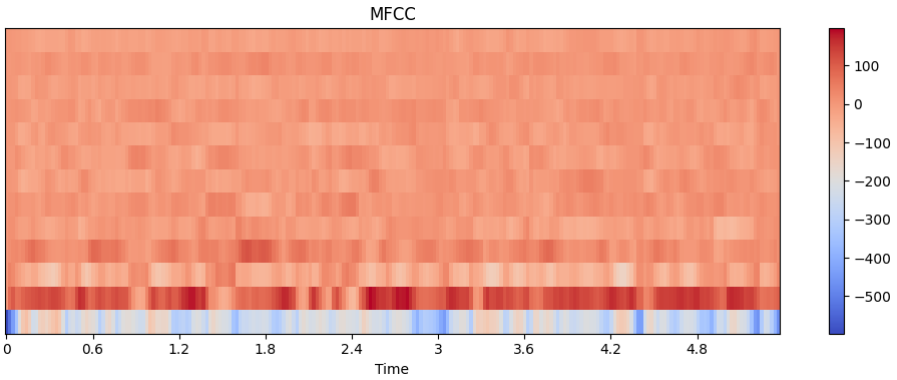


Figure 4.5 : Représentation MFCCs du siganl « narrative1.wav »

4.3.3. Calcul de distance

Pour calculer la distance entre le mot cible et les différents segments du flux à prospecter, nous utilisons la technique de l'alignement temporelle dynamique, et en particulier, l'algorithme DTW précedement présenté.

Le DTW est un algorithme qui permet de mesurer la similarité entre deux séquences temporelles, il est particulièrement utile lorsque les séries temporelles ne sont pas alignées de manière linéaire et peuvent présenter des variations de vitesse ou de délai.

Voici le code standard de l'algorithme DTW de base :

https://www.ionos.fr/digitalguide/sites-internet/developpement-web/dynamic-time-warping/

```
defdtw(s, t, window):
   n, m =len(s),len(t)
   w =np.max([window,abs(n-m)])
dtw_matrix=np.zeros((n+1, m+1))

foriinrange(n+1):
for j inrange(m+1):
dtw_matrix[i, j]= np.inf
dtw_matrix[0,0]=0

foriinrange(1, n+1):
for j inrange(np.max([1,i-w]),np.min([m,i+w])+1):
dtw_matrix[i, j]=0

foriinrange(1, n+1):
for j inrange(np.max([1,i-w]),np.min([m,i+w])+1):
        cost =abs(s[i-1]- t[j-1])
```

```
# take last min from a square box
last_min=np.min([dtw_matrix[i-1, j],dtw_matrix[i, j-1],dtw_matrix[i-1, j-1]])
dtw_matrix[i, j]= cost +last_min
returndtw_matrix
```

Pour accélerer le traitement de comparaison, nous utlisons dans ce travail l'algorithme « fast DTW » qui est une variante optimisée du DTW. La principale idée derrière cette variante est l'élagage par fenêtres en se concentrant sur les points proches de la diagonale. Une implémentation de cet algorithme en Python est disponible via le projet fastdtw.

4.3.4. Décision

Pour chaque signal parole à analyser, nous calculons la distance entre le mot cible et les différents segments obtenus après la segmentation de ce signal. D'abord, on sélectionne le segment qui a obtenu la plus petite distance. Ensuite, cette mesure de distance est comparée à un seuil ; de la valeur de ce seuil dépendra la pertienence de la solution.

Si le score de similarité obtenu est inférieure à ce seuil, le signal de départ est considéré comme contenant le mot clé cible. Dans le cas contraire, le signal est considéré comme ne contenant pas le mot clé cible. Cette méthode nous permet de déterminer la présence ou l'absence du mot clé cible dans chaque segment du flux, en utilisant un seuil personnalisé qui a été optimisé pour notre application spécifique.

4.3.5. Localisation du terme à rechercher

Dans le cas où le signal est jugé contenant le mot clé, le système localise le segment de parole qui correspond à la plus petite distance obtenu précédemment. Le segment étant défini en se basant sur les frontières de début et de fin du segment en question.

4.4. Expérimentation

Pour tester notre proposition, nous considérons les enregistrements mis à disposition par l'association internationale de phonétique IPA, pour l'Arabe et l'Anglais. Le choix des deux langues vise à monter l'indépendance de l'approche proposée de la langue à considérer.

4.4.1. Le dataSet

Les enregistrements des récits que nous considérons sont extraits de la fable « wind and the sun », dont le texte en Anglais est :

The North Wind and the Sun were disputing which was the stronger, when a travelercame along wrapped in a warm cloak.

They agreed that the one who first succeeded in making the traveler take his cloak off shouldbe considered stronger than the other.

Then the North Wind blew as hard as he could, but the more he blew the more closely did the traveler fold his cloak around him;

and at last the North Wind gave up the attempt. Then the Sun shined out warmly, and immediately the traveler took off his cloak.

And so the North Wind was obliged to confess that the Sun was the stronger of the two.

(source : https://en.wikipedia.org/wiki/The_North_Wind_and_the_Sun)

Les enregistrements de l'Anglais et de l'Arabe sont accessible à partir du lien suivant : https://www.internationalphoneticassociation.org/content/ipa-handbook-downloads

Il y a un total de six enregistrements pour l'Anglais et sept pour l'Arabe. Trois mots clés sont à retrouver, il s'agit de « sun », « wind », et « traveller » pour l'Anglais et leurs contreparties en Arabe, à savoir : «شمس » , «ريح», et « مسافر », dont les transcriptions phonétiques sont : [ʃams], [riħ], et [musa:fir]. Les mots clés ont été extraits du fichier « narrative1 ».

La figure 3.5. illustration la position des mots cibles dans le signal considéré en Anglais.

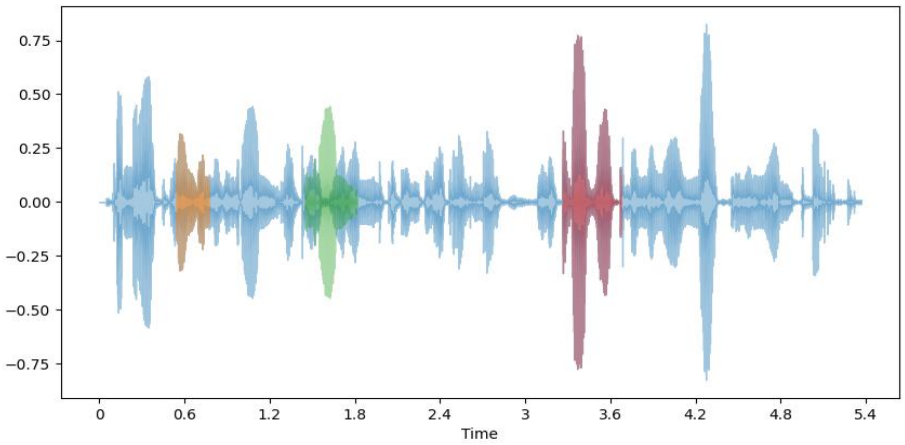


Figure 4.6 : Position des mots cibles dans le signal de départ

Pour la langue arabe, les mots cibles ont été extraits du premier enregistrement, soit « narrative1 », tandis que le 3ème a été extrait du fichier « narrative2 ». Le tableau 1 montre la distribution des mots cibles dans les fichiers du corpus de l'IPA, .

Table 3.1. Distribution des mos cibles dans les différents fichiers à prospecter

	Wind /شمس	Sun / ريح	Traveller / مسافر
Narrative1 (Anglais)	1	1	1
Narrative2 (Anglais)	0	0	1
Narrative3 (Anglais)	1	0	1
Narrative4 (Anglais)	1	0	0
Narrative5 (Anglais)	0	1	1
Narrative6 (Anglais)	1	1	0
Narrative1 (Arabe)	1	1	0
Narrative2 (Arabe)	0	0	1
Narrative3 (Arabe)	0	0	1
Narrative4 (Arabe)	1	0	0
Narrative5 (Arabe)	1	0	1
Narrative6 (Arabe)	0	1	1
Narrative7 (Arabe)	1	1	0

4.4.2. Illustration

En guise d'illustration, nous considérons le fichier « northwind2 » accessible depuis la page ; https://www.mq.edu.au/about/about-the-university/our-faculties/medicine-and-

health-sciences/departments-and-centres/department-of-linguistics/our-research/phonetics-and-phonology/speech/australian-english-pronunciation-and-transcription/the-north-wind-and-the-sun.

Le texte de cet enregistrement est : « when a traveller came along wrapped in a warm cloak». Nous allons y rechercher les trois mots cibles précédemment mentionnés.

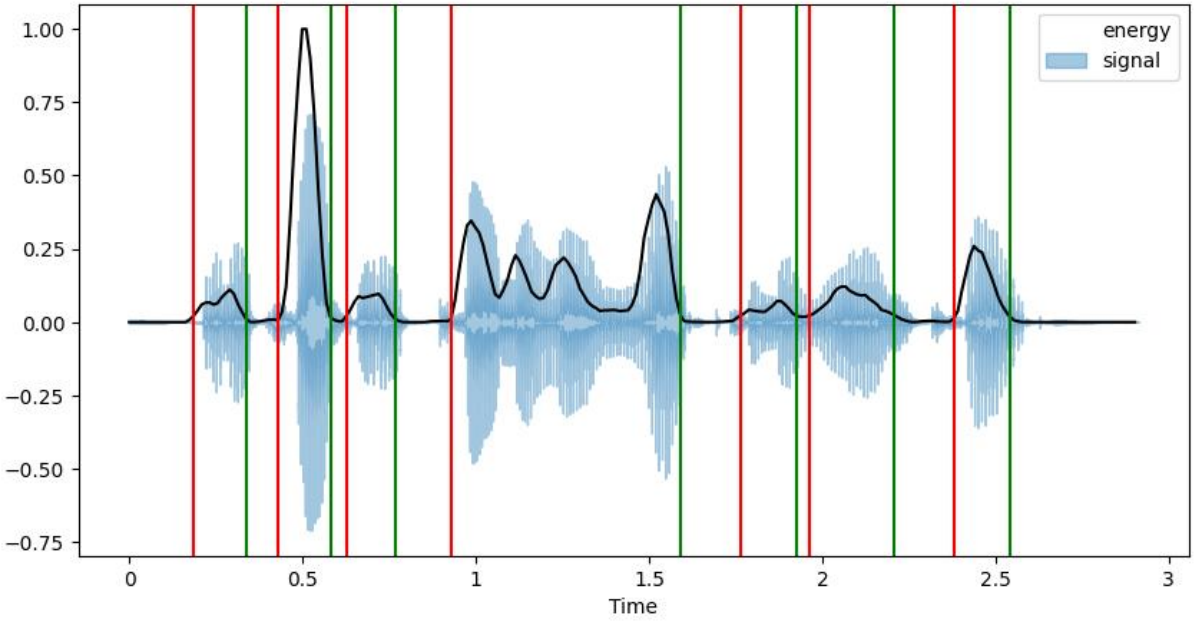


Figure 4.7 : Segmentation du fichier « northwind2 »

4.4.3. Résultats :

Les résultats obtenus pour le seuil de segmentation de 0.25% et le seuil de détection de 500 présentent des performances variables pour les trois mots clés étudiés.

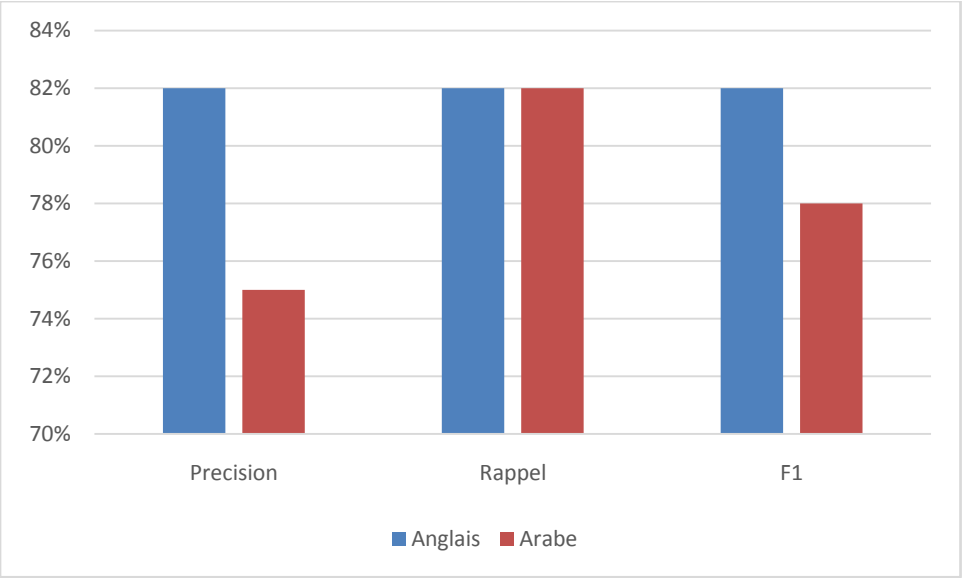


Figure 4.8 : Résultat obtenu en termes de précision, de rappel et du F1

Ces résultats soulignent que la détection des mots clés peut varier en fonction des mots spécifiques et de la complexité de leur reconnaissance dans les flux audio. Il est important de noter que ces performances sont basées sur les seuils de segmentation et de détection choisis, et d'autres combinaisons de seuils pourraient entraîner des résultats différents.

4.5. Conclusion

Ce chapitre détaille notre méthode unique de détection des mots parlés et les résultats que nous avons obtenus sur notre propre ensemble de données. Notre approche s'appuie sur un algorithme qui exploite l'énergie comme critère pour identifier et détecter les segments. Les évaluations que nous avons menées ont montré des performances variées, nous avons rencontré des complications telles que des faux positifs, comme l'a confirmé notre analyse des erreurs. Pour améliorer la précision, nous recommandons d'envisager des modifications des seuils de détection, d'adopter des techniques avancées d'apprentissage automatique et d'élargir la gamme de données d'apprentissage. Le système de détection de mots parlés peut-être amélioreé, à l'avenir, en augmentant potentiellement sa précision et sa robustesse

En somme, ce chapitre constitue une étape cruciale de notre travail, en proposant une méthode non-supervisé pour la détection des mots parlés et en fournissant une évaluation

approfondie de notre modèle sur un dataset spécifiquement conçu. Les résultats obtenus offrent des perspectives prometteuses pour l'amélioration des performances des applications de traitement du langage naturel, ouvrant ainsi la voie à de nombreuses applications pratiques dans des domaines tels que l'assistance vocale, l'indexation automatique de contenus audio et bien plus encore.

Chapitre 5 :Un réseau CNN pour la Détection de Mots parlés

5.1. Introduction

Comme nous venons de le voir dans notre première contribution, l'approche acoustique vise à évaluer la similarité entre la requête parlée et l'énoncé de test, et ce en utilisant un algorithme qui relève de la programmation dynamique, le plus souvent le l'algorithme DTW (Dynamic Time Warping). Toutefois, le DTW ainsi que ses variantes ont été largement critiqués en raison de leur lenteur (Madhavi 2017).

Dans notre deuxième contribution, nous nous affranchissons du calcul de distance de similarité et tirons parti de nouvelles architectures d'apprentissage profond.

Nous suggérons de considérer le problème QbE-STD comme un problème de classification en utilisant l'algorithme des réseau de neurones convolutionnels (CNN) entraîné de manière auto-supervisée pour détecter la présence ou l'absence de la requête parlée inconnue dans un énoncé test à partir d'archives de parole.

À cette fin, un ensemble de données d'apprentissage est construit dans lequel la requête parlée est ajoutée à la fin de l'archive de parole. Lorsque le terme parlé existe initialement dans l'énoncé de test, le nouveau fichier contient au moins deux énoncés de la requête, sinon, la requête parlée existe une fois dans le fichier de parole produit. Au cours de la phase de test, la requête entrante est concaténée à l'énoncé à prospecter et transmise au modèle précédemment entraîné, le modèle étant censé détecter la présence unique ou multiple de la requête parlée.

5.2. Réseaux de neurones convolutifs (CNN) :

Un réseau de neurones convolutifs (CNN) est un type particulier de réseaux de neurones qui a été largement appliqué à une variété de problèmes de reconnaissance de formes, tels que la vision par ordinateur, la reconnaissance vocale , la classification d 'images, la détection d 'objets, la segmentation sémantique etc . ils font partie des architectures de réseaux neuronaux profonds les plus anciennes (Abdel-Hamid et al ., 2014). Ils ont été initialement inspirés par la découverte faite par Hubel et Wiesel , en 1962. Ils peuvent

recevoir n'importe quel type de données en entrée, telles que l'audio, la vidéo, les images, la parole et le langage naturel (Kamilaris et al., 2018). Ainsi, l'architecture des CNN comprend :

5.2.1. Couche de convolution (Convolutional Layer)

C'est la composante clé des réseaux de neurones convolutifs. Son but est de repérer la présence d'un ensemble de caractéristiques (*features)* dans les images reçues en entrée . L'opération de convolution prend deux entrées : une matrice d'image de dimension h×w×d (où h est la hauteur, w est la largeur et d est le nombre de canaux) et un filtre ou un noyau de dimension d. La convolution est réalisée de manière itérative en déplaçant un filtre sur toute la matrice ou l'image d'entrée. À chaque position, la matrice sous le filtre est multipliée par le filtre lui-même, et les produits sont ensuite additionnés pour produire un seul résultat. Ce résultat contribue à former un vecteur de caractéristiques. Ce processus est répété pour chaque position afin de couvrir toutes les parties de l'image (Hamlaoui, 2021).
On obtient pour chaque paire (image, filtre) une carte d'activation, ou ***featuremap***, qui nous indique où se situent les *features* dans l'image: plus la valeur est élevée, plus l'endroit correspondant dans l'image ressemble à la *feature*.

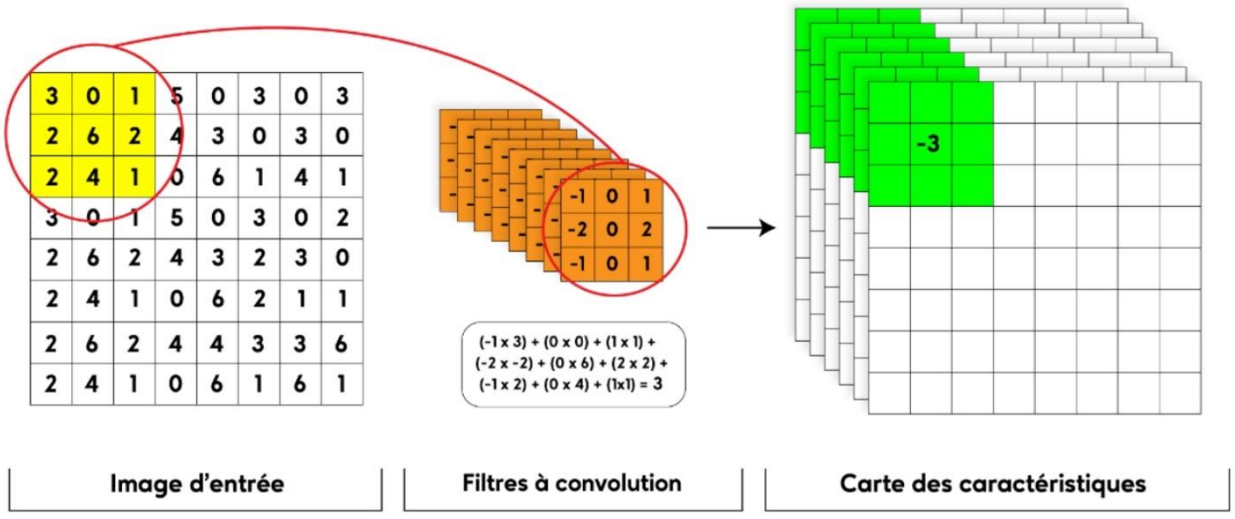


Figure 5.1 :Exemple d'opération de convolution

5.2.2. Couche de Pooling (Pooling layer)

Ce type de couche sert à réduire le nombre de paramètres et la puissance de calcul requise , tout en préservant les caractéristiques les plus importantes. En pratique, on découpe l'image

en cellules régulières espacées les unes des autres d'un stride (pas), afin de ne pas perdre trop d'informations, puis on garde au sein de chaque cellule la valeur maximale. L'image est divisée en petites cellules carrées adjacentes , on obtient en sortie le même nombre de *featuremaps* qu'en entrée, mais celles-ci sont bien plus petites. Il existe deux principaux types de pooling : max-pooling and average-pooling. Max-pooling renvoie la valeur maximale de la partie de l'image couverte par le noyau. Tandis que, average-pooling renvoie la moyenne de toutes les valeurs de la partie de l'image couverte par le noyau. Le pooling permet de contrôler le surapprentissage (overfitting) en réduisant le nombre de paramètres et de calculs dans le réseau (Kpêtchéhoué, 2018).

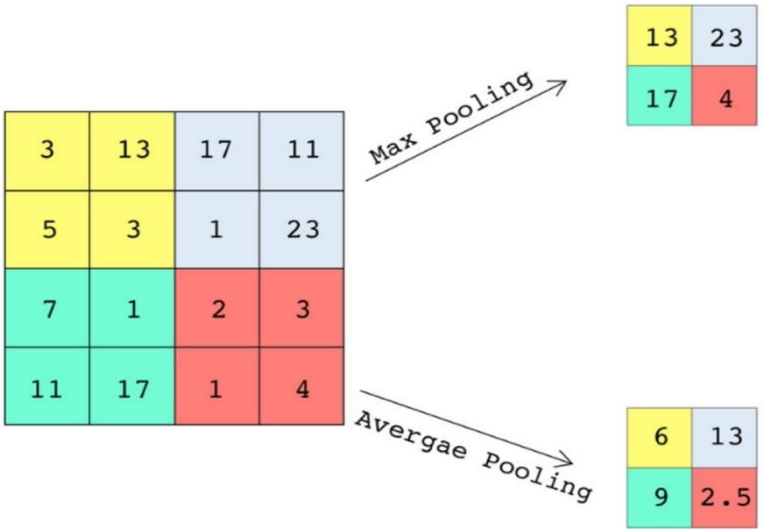


Figure 5.2 : Exemple d'opération de pooling

5.2.3. Couche d'Aplatissement (Flatten Layer)

C'est un élément crucial des réseaux de neurones convolutifs (CNN) qui permet de reconnaître et de classer efficacement une image en vision par ordinateur. Après avoir été traitée par les couches de convolution et de pooling, l'image d'entrée est divisée en caractéristiques et analysée de manière indépendante. Ensuite, la couche entièrement connectée (FC) transforme la sortie de la couche de pooling en une forme adaptée à la classification de l'image. (Tilahun, 2020)

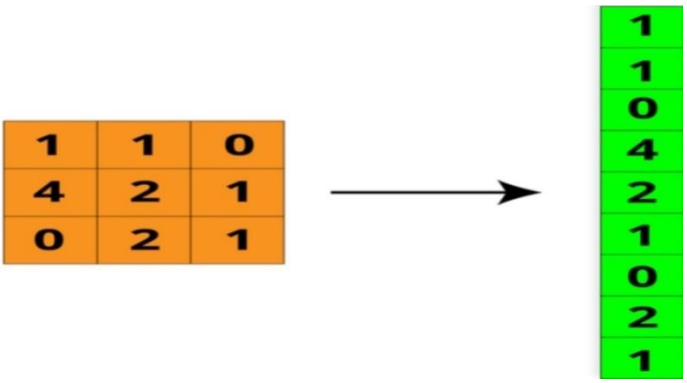


Figure 5.3 : Mécanisme d'aplatissement

5.2.4. Paramètres d'un CNN

5.2.4.1. Filtre

Aussi appelé noyau, le filtre est une petite matrice de poids qui est appliquée à des régions de l'image d'entrée pour extraire des caractéristiques. Par exemple, un filtre 3×3 est une matrice de poids de taille 3 par 3 qui parcourt l'image d'entrée en appliquant des opérations de convolution. Chaque filtre extraie des caractéristiques spécifiques

5.2.4.2. Stride

Le stride (pas) est la longueur de déplacement du filtre à chaque étape lors de son parcours de l'image d'entrée. Un stride de 1 signifie que le filtre se déplace d'un pixel à la fois, tandis qu'un stride de 2 signifie qu'il se déplace de deux pixels à la fois. Un stride plus élevé réduit la taille de la sortie de la convolution, car le filtre se déplace plus rapidement à travers l'image.

Figure 5.4:Illustration du stride

5.2.4.3. Zéro Padding

Le zéro padding consiste à ajouter des zéros autour des bords de l'image d'entrée avant d'appliquer la convolution. Cela permet de conserver la taille de l'image en sortie de la convolution, en particulier lorsque des filtres de grande taille sont utilisés ou lorsque le stride est supérieur à 1. Le zéro padding aide également à éviter la perte d'information sur les bords de l'image, en permettant au filtre de parcourir ces zones.

5.2.4.4. Fonction d'activation ReLU (unité linéaire rectifiée)

La fonction d'activation ReLU est actuellement largement préférée par rapport aux fonctions sigmoïdes et tanh en raison de ses avantages. Elle résout notamment le problème de gradient nul qui peut survenir avec ces fonctions, améliorant ainsi les performances des réseaux de neurones. La plage de valeurs de ReLU s'étend de 0 à l'infini.

5.3. Proposition

La présente contribution représente une alternative aux approches précédentes et s'adapte aux scénarios à faibles ressources car elle ne nécessite pas de connaissances préalables sur le langage cible et évite les calculs de distance à la recherche de patterns. En fait, la proposition exploite les avancées réalisées en vision par ordinateur car nous y considérons les signaux de parole bruts (raw speech) et la détection du terme parlé de la requête est traitée comme un problème de classification d'images binaires.

Étant donné une archive de parole à prospecter, le flux de parole est divisé en segments d'environ quelques secondes, puis chaque segment est étudié pour détecter la présence (ou non) de la requête orale entrante. Lorsque le terme parlé à détecter est fourni au système de détection, l'archive de parole et le terme cible sont concaténés et transmis au système de détection et traités comme une image par un réseau neuronal convolutionnel, en mode End-to-End.

Comme le modèle CNN est par nature un modèle DNN entraîné de manière supervisée, nous suggérons de l'entraîner de manière auto-supervisée en créant un ensemble de données artificiel. L'ensemble de données d'apprentissage est divisé en deux parties pour le rendre équilibré entre les classes positives et négatives.

Le système QbE-STD proposé est construit sur le modèle CNN comme extracteur de caractéristiques et les couches entièrement connectées comme classificateur, entraîné de manière auto-supervisée. La figure 4.1 donne un aperçu du système QbE-STD proposé.

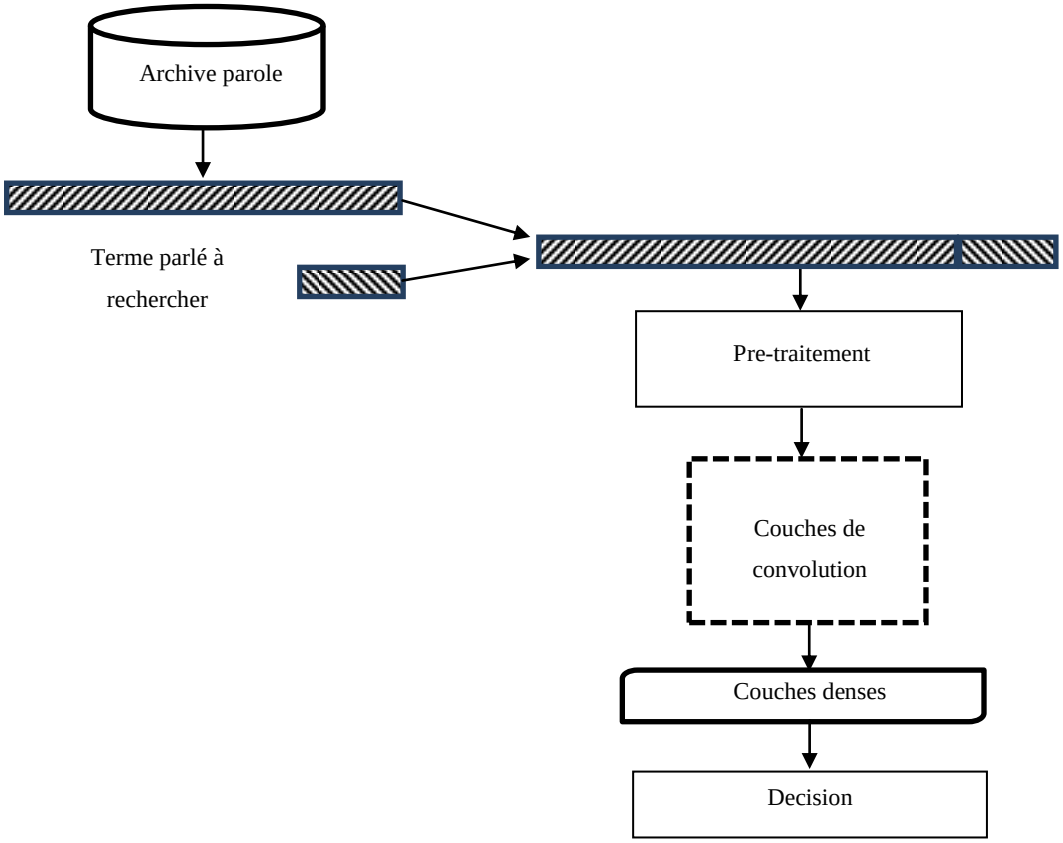


Figure 5.5. Architecture générale du système de QbE-STD

La première partie comprend des segments de parole où la requête parlée existe dans l'archive de parole. De tels échantillons sont construits en ajoutant une requête parlée cible à la fin de l'archive de parole qui comprend déjà une réalisation du terme cible, ainsi, le nouveau segment de parole comprend deux ou plusieurs réalisations du terme parlé cible.

La figure 5.2 illustre la cas où spectrogramme de l'archive contenant une requête parlée est concaténée à une autre réalisation de cette requête. Dans cet exemple, l'énoncé test est illustré par le fichier « narrative1.wav » de l'archive API accessible à l'adresse https://www.internationalphoneticassociation.org/content/ipa-handbook-downloads, son contenu correspond au texte : « The north wind and the sun were disputing which was the stronger, when a traveller came along wrapped in a warm cloak». Alors que la requête parlée correspondant au mot « traveller » est extraite de l'enregistrement « narrative5.wav ». Lorsque les deux signaux sont concaténés, on peut voir que le spectrogramme résultant a deux réalisations du même objet.

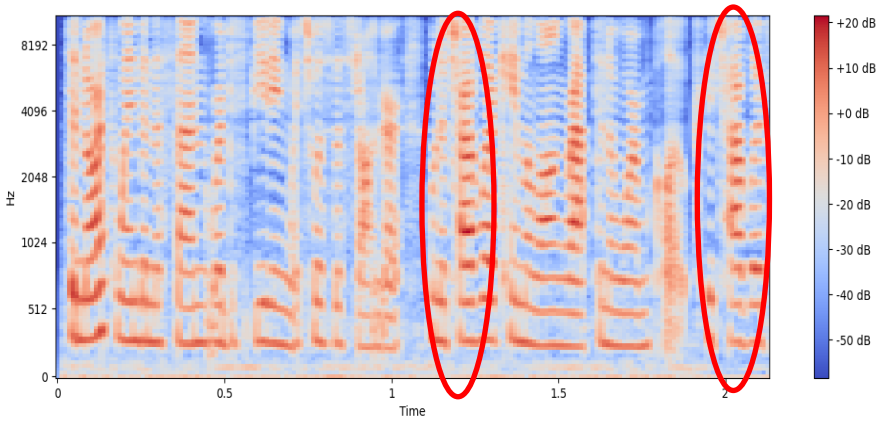


Figure 5. 6 : Illustration par un spectrogramme d'un enregistrement contenant deux réalisations du même terme

La seconde partie de l'ensemble de données contient des segments de discours illustrant le cas où la requête vocale cible n'existe pas dans l'archive vocale. Ici, la requête vocale cible est ajoutée à la fin de l'archive vocale qui ne contient aucune réalisation de ce terme, ce qui conduit à des segments qui n'ont qu'une seule occurrence de la requête.

Le système de détection est entraîné sur l'ensemble de données construit. Au cours de la phase de test, en considérant l'archive à examiner et une requête entrante inconnue, la requête est concaténée à l'archive et transmise au modèle entraîné. Le système est censé détecter si le dernier élément existe déjà dans l'archive ou non.

Par conséquent, l'étape de détection est traitée comme un problème de classification binaire et est implémentée via les couches entièrement connectées.

5.3.1. Représentation du signal

La présence de variations spectrales et de corrélations locales dans les signaux de parole fait des CNN le modèle de DNN le plus approprié pour traiter les applications de traitement de la parole. La parole brute est transmise au modèle CNN pour extraire les caractéristiques pertinentes. Le CNN proposé est composé de quatre blocs consécutifs, chacun comprenant : une couche de convolution, une couche max_pooling et une couche dropout. Cette architecture s'inspire de celle d'AlexNet(Krizhevskyet al., 2012) qui a rencontré un grand succès dans les applications de reconnaissance vocale.

En effet, le nombre de filtres augmente dans les couches les plus profondes du réseau, tandis que la taille des fenêtres diminue lorsque le réseau devient plus profond. En effet, les premières couches capturent des caractéristiques de bas niveau telles que des

motifs simples alors que les couches plus profondes sont destinées à capturer des motifs plus complexes.

5.3.2. La détection de mots parlés

Une fois les caractéristiques extraites de la parole brute, le vecteur acoustique obtenu est transmis aux couches denses suivantes. La dernière couche comporte un neurone représentant les deux alternatives, à savoir : le terme parlé est détecté ou non. Le modèle FC-CNN comprend trois couches denses consécutives.

La fonction « ReLU » a été utilisée comme fonction d'activation pour toutes les couches, à l'exception de la couche de sortie où la fonction « sigmoïde » a été utilisée. Le taux d'apprentissage a été varié de 0.001 à 0,0005 avec l'optimiseur Adam ; et pour éviter le surajustement, une régularisation L1 de 0,001 a été utilisée sur les différentes couches du modèle.

5.4. Experimentation

5.4.1. Dataset

Notre Data-Set est un ensemble de données audio de mots parlés connu sous le nom de "Speech Commands", ce data-set a été créé en 2018 par Google en collaboration avec l'Université de Cambridge, et il a été conçu pour aider à entraîner les systèmes de reconnaissance vocale. Le but était de fournir une base de données suffisamment grande et diversifiée pour entraîner des modèles de reconnaissance vocale pour des commandes vocales simples. Les enregistrements ont été réalisés par des locuteurs de différentes langues, avec différentes intonations et différents accents. Le data-set Speech Commands est devenu une référence dans la communauté de la reconnaissance vocale.

Ce jeu de données comprend 30 mots distincts tels que "backword", "bed", "bird", "cat", "dog", etc. Nous avons sélectionné ces mots pour évaluer notre système de reconnaissance vocale en utilisant un ensemble de trois mots clés, à savoir "right", "left" et "down". Ces trois mots clés seront utilisés pour effectuer notre test.

Pour le traitement des données nous avons utilisé le module os qui nous permet d'interagir avec le système d'exploitation en d'autres mots il nous permet d'accéder à notre data-set, ensuite on utilise la bibliothèque "librosa" du langage python, cette bibliothèque permet l'analyse et la manipulation de données audio. Elle fournit des outils pour charger,

traiter, analyser, transformer et visualiser des données audios en utilisant des techniques de traitement du signal numérique. Et parmi ces technique on a utilisé "librosa.load"pour charger nos fichier audio, après il y a la bibliothèque "Soundfile" qui permet la lecture et l'écriture des fichier audio, et enfin on utilise la fonction "np.zerp " de la bibliothèque "Numpy" elle permet la création de tableau (array) de valeur zéro ou dans notre cas permet la création des flux qui seront constitués de 3 mots comme par exemple le flux1 qui a été générer est composé par les mots "backword", "five" et "down" et bien sûr en faisant la même chose 6 fois avec les mêmes mots pour que l'ordre des mots change.

Et à la fin nous aurons générer 30 flux sur lesquelles on aura besoin pour atteindre notre objectif. Parmi ces 30 flux chaque 6 flux auront les mêmes mots mais chacune aura un ordre de changement différent, ce qui veut dire qu'on aura 5 flux qui auront des mots différents des autres et bien sur les mots clé vont être dans les flux pour s'assurer de la validité du modèle.

5.4.2. Résultats

Les applications ciblées, telles que l'indexation d'archives vocales ou la recherche multimédia, ont tendance à favoriser une détection précise et à éviter les fausses alarmes. Ainsi, le processus d'évaluation est principalement basé sur le taux de détection, le taux de fausses alarmes, la précision et les scores de rappel. La figure 5.7 présente la progression de la précision et de l'erreur durant la phase d'apprentissage, tandis que la précision dans la phase de test a atteint 72.5% (Benati et Bahi (2024)).

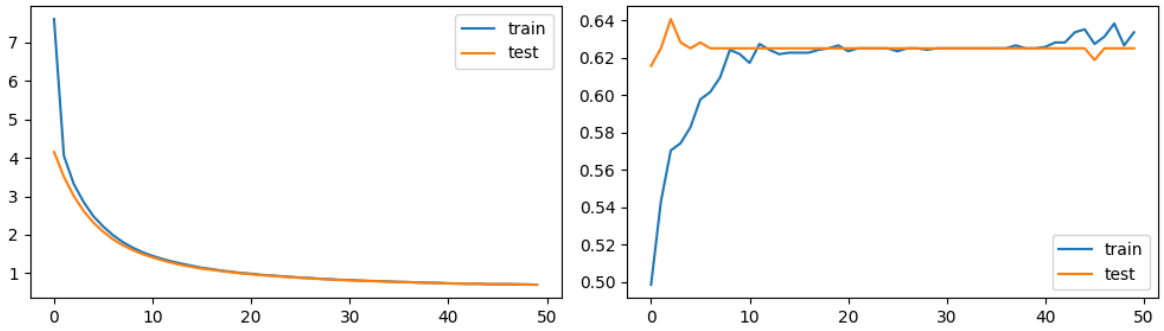


Figure 5.7 : Progression de la précision et de l'erreur durant la phase d'apprentissage

5.4.3. Discussion

Les expériences démontrent le potentiel de l'approche proposée dans un contexte complètement non supervisé. Nous obtenons une précision fort encourageante, en raison de la forte variabilité entre les locuteurs observés pendant la phase d'apprentissage et ceux de la phase de test.

D’autre part, l'étape de prétraitement, qui supprime les silences et accélère ainsi la phase d'apprentissage, affecte également légèrement le modèle.

Conclusion et Perspectives

Tout au long de cette thèse et en utilisant les informations fournies dans les différents chapitres, nous avons exploré une variété d'approches et de méthodes utilisées pour la détecton de termes parlés dans des signaux audio. Pour identifier ces termes dans les flux audio, nous avons réussi à combiner des méthodes spécifiques, notamment la segmentation temporelle, la comparaison de similarité, l'extraction de caractéristiques et l'analyse spectrale. Ensuite, nous avons essayé de tirer profit des avancées offertes par le deep learning pour aller plus avant dans la détection de mots clés qui ne sont pas connus à l'avance de la requête, car nous nous positionnons dans le cadre du QbE-STD.

En ce qui concerne l'approche acoustique, avec une segmentation sur la base de l'énergie du signal, il convient de garder à l'esprit que cette méthode n'est pas infaillible et peut venir avec son propre ensemble d'obstacles. Par exemple, elle peut avoir du mal à se différencier au milieu de beaucoup de bruit, nécessiter un calibrage et un ajustement délicats en fonction des mots clés que vous ciblez et avoir des seuils de détection variables en fonction des circonstances d'enregistrement. De plus, il est démontré que ces méthodes basées sur le calculs de similarité sont gourmandes en temps, bien que le fait de partir sur une segmentation ciblée, permet de réduire cette complexité.

Ensuite, pour l'approche basée sur le deep learning, nous tirons profit des capacités des réseaux de neurones convolutionnels à capturer des caractéristiques discriminantes du signal (la forme) en entrée du réseau pour détecter les mots à rechercher dans le flux à prospecter. Cette méthode nous semble une réelle alternative aux systèmes KWS où le mot à rechercher est connu à l'avance.

En guise de perspectives pour améliorer nos résultats voici quelques pistes d'amélioration à considérer :

D'abord, utiliser des techniques d'amélioration du signal de la parole (speech enhancement) qui sont susceptibles d'améliorer les résultats de la détection lors de la comparaison entre les segments de parole, en supprimant les bruits environnementaux.

L'optimisation des seuils de détection pour l'approche acoustique, au travers d'un tuning laborieux est susceptible d'améliorer les résultats de détection.

Explorer d'autres caractéristique audio, en effet, en plus de l'énergie il y a des

caractéristiques audios pertinentes telles que les MFCC, le spectrogramme ou d'autres descripteurs pour capturer des informations supplémentaires et améliorer la capacité de discrimination du modèle.

Profiter des modèles pré-entrainés dans un cadre de transfer learning pour la partie extraction des caractéristiques du signal.

Tester d'autres architectures pour la partie classification du CNN.

Aller plus en avant dans l'utilisation des mécanismes d'attention dans la modélisation du CNN.

Ces solutions offrent des opportunités d'amélioration et peuvent être explorées dans le cadre de travaux futurs pour renforcer la performance et la fiabilité du système de détection des mots parlés dans un système d'interrogation par l'exemple.

Bibliographie

Ahmad AR, Viard-Gaudin C, Khalid M (2009) “Lexicon-based word recognition using support vector machine and hidden Markov model”. In: International conference on document analysis and recognition (ICDAR’09), pp 161–165

Albert E.-T. , C. Lemnaru, M. Dinsoreanu, and R. Potolea (2019) “Keyword spotting using dynamic time warping and convolutional recurrent networks,” in Proceedings of ICCP – 15th International Conference on Intelligent Computer Communication and Processing, September 5-7, Cluj-Napoca, Romania, pp. 53–60.

Alvarez R. and H.-J. Park (2019) “End-to-end streaming keyword spotting,” in Proceedings of ICASSP 2019 – 44th IEEE International Conference on Acoustics, Speech and Signal Processing, May 12-17, Brighton, UK, pp. 6336–6340.

An. S, Y. Kim, H. Xu, J. Lee, M. Lee, and I. Oh (2019) “Robust keyword spotting via recycle-pooling for mobile game,” in Proceedings of INTERSPEECH – 20th Annual Conference of the International Speech Communication Association, September 15-19, Graz, Austria, pp.3661–3662.

Arik S. O., M. Kliegl, R. Child, J. Hestness, A. Gibiansky, C. Fougner, R. Prenger, and A. Coates (2017) “Convolutional recurrent neural networks for small-footprint keyword spotting,” in Proceedings of INTERSPEECH – 18th Annual Conference of the International Speech Communication Association, August 20-24, Stockholm, Sweden, pp. 1606–1610.

Audhkhasi K, Verma A (2007) “Keyword spotting using modified minimum edit distance measure”. In: Proceedings of ICASSP, vol 4, pp 929–932

Ayed YB, Fohr D, Haton JP, Chollet G (2002) “Keyword spotting using support vector machines”. In: International conference on text, speech and dialogue, pp 285–292

Bahi H, Benati N (2009) “A new keyword spotting approach. In: International conference on multimedia computing and systems (ICMCS’09), pp 77–80

Bai Y., J. Yi, H. Ni, Z. Wen, B. Liu, Y. Li, and J. Tao (2016) “End-to-end keywords spotting based on connectionist temporal classification for Mandarin,” in Proceedings of ISCSLP – 10th InternationalSymposium on Chinese Spoken Language Processing, October 17-20, Tianjin, China.

Bai Y., J. Yi, J. Tao, Z. Wen, Z. Tian, C. Zhao, and C. Fan (2019)“A time delay neu ral network with shared weight self-attention for small-footprint keyword spotting,” in Proceedings of INTERSPEECH – 20th Annual Conference of the International Speech Communication Association, September 15-19, Graz, Austria, 2019, pp. 2190–2194.

Bazzi I (2002) “Modelling out-of-vocabulary words for robust speech recognition”. Massachusetts Institute of Technology, Cambridge

Benati N, Bahi H (2016) "Spoken term detection based on acoustic speech segmentation," *2016* 7th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), Hammamet, Tunisia, 2016, pp. 267-271. doi: 10.1109/SETIT.2016.7939878

Benati N, Bahi H (2024). Self-supervised spoken term detection for query by example. Ingénierie des Systèmes d’Information, Vol. 29, No. 3, pp. 1175-1181. https://doi.org/10.18280/isi.290334

Benayed Y, Fohr D, Haton JP, Chollet G (2003) “Improving the performance of a keyword spotting system by using support vector machines”. In: IEEE workshop on automatic speech recognition and understanding (ASRU’03), pp 145–149

Berg A., M. O’Connor, and M. T. Cruz (2021) “Keyword Transformer: A self-attention model for keyword spotting,” in Proceedings of INTERSPEECH – 22nd Annual Conference of the International Speech Communication Association, August 30-September 3, Brno, Czechia, pp. 4249–4253.

Bishop, C. M., Nasrabadi, N. M. (2006). Pattern recognition and machine learning (Vol. 4, No. 4, p. 738). New York: springer.

Bridle JS (1973) “An efficient elastic-template method for detecting given words in running speech”. In: British Acoustical Society meeting, pp 1–4

Bridle J. S. (1990) “Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition,”Neurocomputing, pp. 227–236.

Burget L et al. (2008) “Combination of strongly and weakly constrained recognizers for reliable detectionof OOVs”. In: International conference on acoustics, speech and signal processing (ICASSP’08), pp 4081–4084

Cernocky J et al. (2007) “Search in speech for public security and defense”. In: IEEE workshop on signal processing applications for public security and forensics (SAFE), pp 1–7.

Chang, E. I., Lippmann, R. P. (1996). Improving wordspotting performance with artificially generated data. In 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings (Vol. 1, pp. 526-529).

Ceolini E., J. Anumula, S. Braun, and S.-C. Liu (2019) “Event-driven pipeline for low-latency low-compute

Chai S., Z. Yang, C. Lv, and W.-Q. Zhang (2019) “An end-to-end model based on TDNN-BiGRU for keyword spotting,” in Proceedings of IALP –International Conference on Asian Language Processing, November 15-17, Shanghai, China, 2019, pp. 402–406.

Chen CP, Bilmes JA (2007) “MVA processing of speech features”. IEEE Trans Audio Speech Lang Process 15:257–270

Chen G., C. Parada, and G. Heigold (2014) “Small-footprint keyword spotting using deep neural networks,” in Proceedings of ICASSP 2014 – 39th IEEE International Conference on Acoustics, Speech and Signal Processing, May 4-9, Florence, Italy, pp. 4087–4091.

Chen G., C. Parada, and T. N. Sainath (2015) “Query-by-example keyword spotting using long short-term memory networks,” in Proceedings of ICASSP – 40th IEEE International Conference on Acoustics, Speech and Signal Processing, April 19-24,

Brisbane, Australia, 2015, pp. 5236–5240.

Chen X., S. Yin, D. Song, P. Ouyang, L. Liu, and S. Wei (2019) “Small-footprint keyword spotting with graph convolutional network,” in Proceedings of ASRU – IEEE Automatic Speech Recognition and Understanding Workshop, December 14-18, Singapore, Singapore, pp. 539–546.

Chen Y., T. Ko, L. Shang, X. Chen, X. Jiang, and Q. Li (2020) “An investigation of few-shot learning in spoken term classification,” in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication Association, October 25-29, Shanghai, China, pp. 2582–2586.

Choi S., S. Seo, B. Shin, H. Byun, M. Kersner, B. Kim, D. Kim, and S. Ha (2019) “Temporal convolution for real-time keyword spotting on mobile devices,” in Proceedings of INTERSPEECH – 20th Annual Conference of the International Speech Communication Association, September 15-19, Graz, Austria, pp. 3372–3376.

Christiansen, R., Rushforth, C. (1977). Detecting and locating key words in continuous speech using linear predictive coding. IEEE Transactions on Acoustics, Speech, and Signal Processing, 25(5), 361-367.

Chung Y.-A., C.-C. Wu, C.-H. Shen, H.-Y. Lee, and L.-S. Lee (2016) “Audio Word2Vec: Unsupervised learning of audio segment representations using sequence-to-sequence autoencoder,” in Proceedings of INTERSPEECH – 17th Annual Conference of the International Speech Communication Association, September 8-12, San Francisco, USA, pp. 765–769.

Coucke A., M. Chlieh, T. Gisselbrecht, D. Leroy, M. Poumeyrol, and T. Lavril (2019) “Efficient keyword spotting using dilated convolutions and gating,” in Proceedings of ICASSP – 44th IEEE International Conference on Acoustics, Speech and Signal Processing, May 12-17, Brighton, UK, pp. 6351–6355.

de Andrade D. C. , S. Leo, M. L. D. S. Viana, and C. Bernkopf (2018) “A neural attention model for speech command recognition” , arXiv:1808.08929v1.

Du X., M. Zhu, M. Chai, and X. Shi (2018) “End to end model for keyword spotting

with trainable window function and Densenet," in Proceedings of DSP – 23rd IEEE International Conference on Digital Signal Processing, November 19-21, Shanghai, China.

Dumpala SH, Raju Alluri KNRK, Suryakanth VG, Uppala AKV (2015) "Analysis of constraints on segmental DTW for the task of query-by-example spoken term detection". In: Annual IEEE India conference (INDICON), New Delhi, pp 1–6

El Méliani, R., O'Shaughnessy, D. D. (1995). Lexical fillers for task-independent-training based keyword spotting and detection of new words. Eurospeech.

Fernández S., A. Graves, and J. Schmidhuber (2007) "An application of recurrent neural networks to discriminative keyword spotting," in Proceedings of ICANN – 17th International Conference on Artificial Neural Networks, September 9-13, Porto, Portugal, pp. 220–229.

Ferrer L, Estienne C (2001) "Improving performance of a keyword spotting system by using a new confidence measure". In: INTERSPEECH, pp 2561–2564.

Fuchs T. and J. Keshet (2017) "Spoken term detection automatically adjusted for a given threshold," IEEE Journal of Selected Topics in Signal Processing, vol. 11, pp. 1310–1317.

Gao Y., N. D. Stein, C.-C. Kao, Y. Cai, M. Sun, T. Zhang, and S. Vitaladevuni (2020) "On front-end gain invariant modeling for wake word spotting," in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication Association, October 25-29, Shanghai, China, pp. 991–995.

Goodfellow I., Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016, http://www.deeplearningbook.org.

Graves A., S. Fernández, F. Gomez, and J. Schmidhuber (2006) "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in Proceedings of ICML – 23rd International Conference on Machine Learning, June 25-29, Pittsburgh, USA, pp. 369–376.

Hazen TJ, Shen W, White C (2009) "Query-by-example spoken term detection using

phonetic posteriorgram templates". In: Proceedings of IEEE workshop on automatic speech recognition & understanding (ASRU), pp 421–426

He K., X. Zhang, S. Ren, and J. Sun (2016) "Deep residual learning for image recognition," in Proceedings of CVPR – Conference on Computer Vision and Pattern Recognition, June 26-July 1, Las Vegas, USA, , pp. 770–778.

He Y., R. Prabhavalkar, K. Rao, W. Li, A. Bakhtin, and I. Mc-Graw (2017) "Streaming small-footprint keyword spotting using sequence-to-sequence models," in Proceedings of ASRU – IEEE Automatic Speech Recognition and Understanding Workshop, December 16-20, Okinawa, Japan, pp. 474–481.

Hermansky H, Morgan N (1994) "RASTA processing of speech". IEEE Trans Speech Audio Process 2:578–589

Hermansky H, Morgan N, Bayya A, Kohn P (1991) "Compensation for the effect of the communication channel in auditory-like analysis of speech (RASTA-PLP)". In: European conference on speech communication and technology (EuroSpeech), pp 1367–1370

Higgins, A., Wohlford, R. (1985). Keyword recognition using template concatenation. In ICASSP'85. IEEE International Conference on Acoustics, Speech, and Signal Processing (Vol. 10, pp. 1233-1236). IEEE.

Higuchi T., M. Ghasemzadeh, K. You, and C. Dhir (2020) "Stacked 1D convolutional networks for end-to-end small footprint voice trigger detection," in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication Association, October 25-29, Shanghai, China, pp. 2592–2596.

Hochreiter S. and J. Schmidhuber (1997) "Long short-term memory," Neural Computation, vol. 9, pp. 1735–1780.

Hori T, Hetherington IL, Hazen TJ, Glass JR (2007) "Open-vocabulary spoken utterance retrieval using confusion networks". In: Proceedings of ICASSP, pp 73–76.

https://doi.org/10.1109/ SPICES.2015.7091361

Hou J., L. Xie, and Z. Fu (2016) “Investigating neural network based query-by-example keyword spotting approach for personalized wake-up word detection in Mandarin Chinese,” in Proceedings of ISCSLP – 10th International Symposium on Chinese Spoken Language Processing, October 17-20, Tianjin, China.

Hou J., Y. Shi, M. Ostendorf, M.-Y. Hwang, and L. Xie (2020) “Mining effective negative training samples for keyword spotting,” in Proceedings of ICASSP – 45th IEEE International Conference on Acoustics, Speech and Signal Processing, May 4-8, Barcelona, Spain, 2020, pp.7444–7448.

Hou J., Y. Shi, M. Ostendorf, M.-Y. Hwang, and L. Xie(2019) “Region proposal network based small-footprint keyword spotting,” IEEE Signal Processing Letters, vol. 26, pp. 1471–1475.

Howard A. G. , M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam (2017) “MobileNets: Efficient convolutional neural networks for mobile vision applications,” arXiv:1704.04861v1.

Huang J., W. Gharbieh, H. S. Shim, and E. Kim (2021) “Query-by-example keyword spotting system using multi-head attention and soft-triple loss,” in Proceedings of ICASSP – 46th IEEE International Conference on Acoustics, Speech and Signal Processing, June 6-11, Toronto, Canada, pp. 6858–6862.

Huang Y. A., T. Z. Shabestary, and A. Gruenstein (2019) “Hotword Cleaner: Dual-microphone adaptive noise cancellation with deferred filter coefficients for robust keyword spotting,” in Proceedings of ICASSP – 44th IEEE International Conference on Acoustics, Speech and Signal Processing, May 12-17, Brighton, UK, pp. 6346–6350.

Huang Y., T. Hughes, T. Z. Shabestary, and T. Applebaum (2018) “Supervised noise reduction for multichannel keyword spotting,” in Proceedings of ICASSP – 43rd IEEE International Conference on Acoustics, Speech and Signal Processing, April 15-20, Calgary, Canada, pp. 5474–5478.

Huang X, Acero A, Hon H-W (2001) “Spoken language processing: a guide to

theory, algorithm, and system development". Prentice Hall PTR, Upper Saddle River

Ibrahim E. A., J. Huisken, H. Fatemi, and J. P. de Gyvez (2019) "Keyword spotting using time-domain features in a temporal convolutional network," in Proceedings of DSD – 22nd Euromicro Conference on Digital System Design, August 28-30, Kallithea, Greece, pp. 313–319.

Irino T. and M. Unoki (1999) "An analysis/synthesis auditory filterbank based on an IIR implementation of the gammachirp," Journal of the Acoustical Society of Japan, vol. 20, pp. 397–406.

Ji X., M. Yu, J. Chen, J. Zheng, D. Su, and D. Yu (2020) "Integration of multi-look beamformers for multi-channel keyword spotting," in Proceedings of ICASSP – 45th IEEE International Conference on Acoustics, Speech and Signal Processing, May 4-8, Barcelona, Spain, pp. 7464–7468.

Karthik Pandia DS, Saranya MS, Murthy HA (2016) "A fast query-by-example spoken term detection for zero resource languages". In: IEEE SPCOM'16, pp 1–5

Keshet J, Bengio S (2009) "Automatic speech and speaker recognition: large margin and kernel methods". Wiley, London

Keshet J, Grangier D, Bengio S (2009) "Discriminative keyword spotting". Speech Commun 51:317–329

Kiefer. J and J. Wolfowitz (1952) "Stochastic estimation of the maximum of a regression function," The Annals of Mathematical Statistics, vol. 23, pp.462–466.

Kim B., S. Chang, J. Lee, and D. Sung (2021) "Broadcasted residual learning for efficient keyword spotting," in Proceedings of INTERSPEECH – 22nd Annual Conference of the International Speech Communication Association, August 30-September 3, Brno, Czechia, pp. 4538–4542.

Kingma D. P. and J. Ba (2015) "Adam: A method for stochastic optimization," in Proceedings of ICLR 2015 – 3rd International Conference on Learning Representations, May 7-9, San Diego, USA.

Krizhevsky, A., Sutskever, I., Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. Communications of the ACM, 60: 84-90. https://doi.org/10.1145/3065386

Kumar R., V. Yeruva, and S. Ganapathy (2018) "On convolutional LSTM modeling for joint wake-word detection and text dependent speaker verification," in Proceedings of INTERSPEECH – 19th Annual Conference of the International Speech Communication Association, September 2-6, Hyderabad, India, pp. 1121–1125.

LeCun Y. A., L. Bottou, G. B. Orr, and K.-R. Müller (2012) "Efficient BackProp," in Neural Networks: Tricks of the Trade. Springer, vol. 7700, pp. 9–48.

Lee A, Shikano K, Kawahara T (2004) "Real-time word confidence scoring using local posterior probabilities on tree trellis search". In: International conference on acoustics, speech, and signal processing (ICASSP'04), vol 791, pp I-793–796

Lee M., J. Lee, H. J. Jang, B. Kim, W. Chang, and K. Hwang (2019) "Orthogonality constrained multi-head attention for keyword spotting," in Proceedings of ASRU – IEEE Automatic Speech Recognition and Understanding Workshop, December 14-18, Singapore, Singapore, pp. 86–92.

Levin K. , K. Henry, A. Jansen, and K. Livescu (2013) "Fixed-dimensional acoustic embeddings of variable-length segments in low-resource settings," in Proceedings of ASRU – IEEE Automatic Speech Recognition and Understanding Workshop, December 8-12, Olomouc, Czech Republic, pp. 410–415.

Li J, Deng L, Gong Y, Haeb-Umbach R (2014) "An overview of noise-robust automatic speech recognition". IEEE/ACM Trans Audio Speech Lang Process 22:745–777

Li J, Deng L, Gong Y, Haeb-Umbach R (2014) An overview of noise-robust automatic speech recognition. IEEE/ACM Trans Audio Speech Lang Process 22:745–777

Li X., X. Wei, and X. Qin (2020) "Small-footprint keyword spotting with multiscale temporal convolution," in Proceedings of INTERSPEECH – 21st Annual Conference of

the International Speech Communication Association, October 25-29, Shanghai, China, pp. 1987–1991.

Lin H, Bilmes J, Vergyri D, Kirchhoff K (2007) “OOV detection by joint word/phone lattice alignment”. In:IEEE workshop on automatic speech recognition & understanding, (ASRU), pp 478–483

Lin H, Stupakov A, Bilmes J (2008) “Spoken keyword spotting via multi-lattice alignment”. In: INTERSPEECH, pp 2191–2194

Lippmann, R. P., Chang, E. I., Jankowski, C. R. (1994). Wordspotter training using figure-of-merit back propagation. ICASSP'94. IEEE International Conference on Acoustics, Speech and Signal Processing (Vol. 1, pp. I-389).

Liu B., S. Nie, Y. Zhang, S. Liang, Z. Yang, and W. Liu (2019a) “Focal loss and double-edge-triggered detector for robust small-footprint keyword spotting,” in Proceedings of ICASSP – 44th IEEE International Conference on Acoustics, Speech and Signal Processing, May 12-17, Brighton, UK pp. 6361–6365.

Liu B., Y. Sun, and B. Liu (2019) “Translational bit-by-bit multi-bit quantizationfor CRNN on keyword spotting,” in Proceedings of CyberC –International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, October 17-19, Guilin, China, pp. 444–451.

Liu H. , A. Abhyankar, Y. Mishchenko, T. Sénéchal, G. Fu, B. Kulis, N. Stein, A. Shah, and S. N. P. Vitaladevuni (2020) “Metadata-aware end-to-end keyword spotting,” in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication Association, October 25-29, Shanghai, China, pp. 2282–2286.

Liu Z., T. Li, and P. Zhang (2021) “RNN-T based open-vocabulary keyword spotting in Mandarin with multi-level detection,” in Proceedings of ICASSP – 46th IEEE International Conference on Acoustics, Speech and Signal Processing, June 6-11, Toronto, Canada, pp. 5649–5653.

López-Espejo I. (2017) “Robust speech recognition on intelligent mobile devices

with dual-microphone," Ph.D. dissertation, University of Granada.

López-Espejo I., Z.-H. Tan, and J. Jensen (2019) "Keyword spotting for hearing assistive devices robust to external speakers," in Proceedings of INTERSPEECH – 20th Annual Conference of the International Speech Communication Association, September 15-19, Graz, Austria, pp. 3223–3227.

López-Espejo I., Z.-H. Tan, and J. Jensen (2020a) "Exploring filterbank learning for keyword spotting," in Proceedings of EUSIPCO – 28th European Signal Processing Conference, January 18-21, Amsterdam, Netherlands, pp. 331–335.

López-Espejo I., Z.-H. Tan, and J. Jensen (2020b), "Improved external speaker-robust keyword spotting for hearing assistive devices," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 1233–1247.

López-Espejo I., Z.-H. Tan, and J. Jensen (2021) "A novel loss function and training strategy for noise-robust keyword spotting," IEEE/ACM Transactions on Audio, Speech, and Language Processing.

López-Espejo I., Z. -H. Tan, J. H. L. Hansen and J. Jensen (2022) "Deep Spoken Keyword Spotting: An Overview," in IEEE Access, vol. 10, pp. 4169-4199, doi: 10.1109/ACCESS.2021.3139508

Lugosch L. and S. Myer (2018) "DONUT: CTC-based query-by-example keyword spotting," in Proceedings of NIPS 2018 – 32nd Annual Conference on Neural Information Processing Systems, December 2-8, Montreal, Canada, pp. 1–9.

Luo J. Wang J., Cheng N., Tang H., Xiao J. (2022). Speech Augmentation Based Unsupervised Learning for Keyword Spotting," 2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy, 2022, pp. 1-7, doi:10.1109/IJCNN55064.2022.9892207

Madhavi MC, Patil H (2017) "Partial matching and search space reduction for QbE-STD". Comput Speech Lang 45:58–82

Madikeri SR, Murthy HA (2012) "Acoustic segmentation using group delay

functions and its relevance to spoken keyword spotting". In: Text speech and dialogue. Springer, Heidelberg, pp 496–504

Mantena G, Achanta S, Prahallad K (2014) "Query-by-example spoken term detection using frequency domain linear prediction and non-segmental dynamic time warping". IEEE/ACM Trans Audio Speech Lang Process 22(5):946–955

Mary L. and Deekshitha G (2019) "Searching Speech Databases :Features, Techniques and Evaluation Measures", book, SpringerBriefs in Speech Technology, ISBN 978-3-319-97760-7 ; ISBN 978-3-319-97761-4 (eBook), https://doi.org/10.1007/978-3-319-97761-4

Mazzawi H., X. Gonzalvo, A. Kracun, P. Sridhar, N. Subrahmanya, I. L. Moreno, H. J. Park, and P. Violette(2019) "Improving keyword spotting and language identification via neural architecture search at scale," in Proceedings of INTERSPEECH – 20th Annual Conference of the International Speech Communication Association, September 15-19, Graz, Austria, pp. 1278–1282.

Menon R., H. Kamper, J. Quinn, and T. Niesler(2018) "Fast ASR-free and almost zero-resource keyword spotting using DTW and CNNs for humanitarian monitoring," in Proceedings of INTERSPEECH – 19th Annual Conference of the International Speech Communication Association, September 2-6, Hyderabad, India, pp. 2608–2612.

Miller DR et al. (2007) "Rapid and accurate spoken term detection". In: Annual conference of the internationa speech communication association (INTERSPEECH), pp 314–317

Mittermaier S., L. Kürzinger, B. Waschneck, and G. Rigoll (2020) "Small-footprint keyword spotting on raw audio data with sinc-convolutions," in Proceedings of ICASSP – 45th IEEE International Conferenceon Acoustics, Speech and Signal Processing, May 4-8, Barcelona, Spain,pp. 7454–7458.

Mo T. , Y. Yu, M. Salameh, D. Niu, and S. Jui (2020) "Neural architecture search for keyword spotting," in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication

Association, October 25-29, Shanghai, China, pp. 1982–1986.

Molau S, Hilger F, Ney H (2003) “Feature space normalization in adverse acoustic conditions”. In: International conference on acoustics, speech, and signal processing (ICASSP’03), pp I-656–I-659

Morgan, D. P., Scofield, C. L., Adcock, J. E. (1991). Multiple neural network topologies applied to keyword spotting. ICASSP 91: 1991 International Conference on Acoustics, Speech, and Signal Processing (pp. 313-316).

Mostafa H. (2018) “Supervised learning based on temporal coding in spiking neural networks,” IEEE Transactions on Neural Networks and Learning Systems, vol. 29, pp. 3227–3235.

Motlicek P, Valente F, Szoke I (2012) “Improving acoustic based keyword spotting using LVCSR lattices”. In: 2012 IEEE international conference on acoustics, speech and signal processing (ICASSP), IEEE, pp 4413–4416

Muhsinzoda M., C. C. Corona, D. A. Pelta, and J. L. Verdegay (2019) “Activating accessible pedestrian signals by voice using keyword spotting systems,” in Proceedings of ISC2 – IEEE International Smart CitiesConference, October 14-17, Casablanca, Morocco, pp. 531–534.

Myer S. and V. S. Tomar (2018) “Efficient keyword spotting using time delay neural networks,” in Proceedings of INTERSPEECH – 19th Annual Conference of the International Speech Communication Association, September 2-6, Hyderabad, India, pp. 1264–1268.

Nakkiran P., R. Alvarez, R. Prabhavalkar, and C. Parada (2015) “Compressing deep neural networks using a rank-constrained topology,” in Proceedings of INTERSPEECH – 16th Annual Conference of the International Speech Communication Association, September 6-10, Dresden, Germany, pp. 1473–1477.

Ngo K, Spriet A, Moonen M, Wouters J, Jensen SH (2012) “A combined multi-channel Wiener filter-based noise reduction and dynamic range compression in hearing aids”. Sig Process 92:417–426

Obara M, Moriya M, Konno R, Kojima K, Tanaka K, Lee S, Itoh Y (2017) “Acceleration for query-by-example using posteriorgram of deep neural network”. In: Proceedings of APSIPA ASC, Kuala Lumpur, pp 1565–1569

Ou Z, Luo H (2012) “CRF-based confidence measures of recognized candidates for lattice-based audio indexing. In: 2012 IEEE international conference on acoustics, speech and signal processing (ICASSP), IEEE, pp 4933–4936

Parada C., A. Sethy, and B. Ramabhadran (2009) “Query-by-example spoken term detection for OOV terms,” in Proceedings of ASRU – IEEE Automatic Speech Recognition and Understanding Workshop, December 13-17, Moreno, Italy, pp. 404–409.

Park H.-J., P. Violette, and N. Subrahmanya (2020) “Learning to detect keyword parts and whole by smoothed max pooling,” in Proceedings of ICASSP – 45th IEEE Internation al Conference on Acoustics, Speech and Signal Processing, May 4-8, Barcelona, Spain, pp. 7899–7903.

Pattanayak B., J. K. Rout, and G. Pradhan (2019) “Adaptive spectral smoothening for development of robust keyword spotting system,” IET Signal Processing, vol. 13, pp. 544–550.

Pedroni B. U., S. Sheik, H. Mostafa, S. Paul, C. Augustine, and G. Cauwenberghs (2018) “Small-footprint spiking neural networks for power-efficient keyword spotting,” in Proceedings of BioCAS – IEEE Biomedical Circuits and Systems Conference, October 17-19, Cleveland, USA.

Pinto J, Szoke I, Prasanna SRM, Hermansky H (2008) “Fast automatic spoken term detection from sequence of phonemes”. In: Proceedings of 31st annual international ACM SIGIR’08 conference, pp 28–33

Prabhavalkar R., R. Alvarez, C. Parada, P. Nakkiran, and T. N. Sainath, (2015) “Automatic gain control and multi-style training for robust small-footprint keyword spotting with deep neural networks,” in Proceedings of ICASSP– 40th IEEE International Conference on Acoustics, Speech and Signal Processing, April 19-24, Brisbane, Australia, 2015, pp. 4704–4708.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, 77(2), 257-286.

Ram D, Asaei A, Bourlard H (2018) “Sparse subspace modeling for query by example spoken term detection”. In: IEEE/ACM transactions on audio, speech, and language processing, vol 26, no 6, pp 1130–1143

Ramabhadran B, Sethy A, Mamou J, Kingsbury B, Chaudhari U (2009) “Fast decoding for open vocabulary spoken term detection”. In: Proceedings of human language technologies: the 2009 annual conference of the North American Chapter of the Association for Computational Linguistics, companion, volume: short papers, Association for Computational Linguistics, pp 277–280

Rastrow A, Sethy A, Ramabhadran B (2009) “A new method for OOV detection using hybrid word/fragment system”. In: 2009 IEEE international conference on acoustics, speech and signal processing, IEEE, pp 3953–3956

Ravanelli M. and Y. Bengio (2018) “Speaker recognition from raw waveform with SincNet,” in Proceedings of SLT – IEEE Spoken Language Technology Workshop, December 18-21, Athens, Greece, pp. 1021–1028.

Riviello A. and J.-P. David (2019) “Binary speech features for keyword spotting tasks,” in Proceedings of INTERSPEECH – 20th Annual Conference of the International Speech Communication Association, September 15- 19, Graz, Austria, pp. 3460–3464.

Rohlicek, J. R., Jeanrenaud, P., Ng, K., Gish, H., Musicus, B., Siu, M. (1993). Phonetic training and language modeling for word spotting. In 1993 IEEE International Conference on Acoustics, Speech, and Signal Processing (Vol. 2, pp. 459-462).

Rose, R. C., Paul, D. B. (1990). A hidden Markov model based keyword recognition system. In International Conference on Acoustics, Speech, and Signal Processing (pp. 129-132).

Rose, R. C. (1992). Discriminant wordspotting techniques for rejecting non-vocabulary utterances in unconstrained speech. ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing (Vol. 2, pp. 105-108).

Rumelhart D. E. , G. E. Hinton, and R. J. Williams (1986) “Learning representations by back-propagating errors,” Nature, vol. 323, pp. 533–536.

Rybakov O., N. Kononenko, N. Subrahmanya, M. Visontai, and S. Laurenzo (2020) “Streaming keyword spotting on mobile devices,” in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication Association, October 25-29, Shanghai, China, pp. 2277–2281.

Sacchi N., A. Nanchen, M. Jaggi, and M. Cernak (2019) “Open-vocabulary keyword spotting with audio and text embeddings,” in Proceedings of INTERSPEECH – 20th Annual Conference of the International Speech Communication Association, September 15-19, Graz, Austria, pp. 3362–3366.

Sainath T. N. and C. Parada (2015) “Convolutional neural networks for small-footprint keyword spotting,” in Proceedings of INTERSPEECH Association, September 6-10, Dresden, Germany, pp. 1478–1482.

Scherer , A. Müller, and S. Behnke (2010) “Evaluation of pooling operationsin convolutional architectures for object recognition,” in Proceedings of ICANN – 20th International Conference on Artificial Neural Networks, September 15-18, Thessaloniki, Greece, pp. 92–101.

Shan C., J. Zhang, Y. Wang, and L. Xie (2018) “Attention-based end-to-end models for small-footprint keyword spotting,” in Proceedings of INTERSPEECH – 19th Annual Conference of the International Speech Communication Association, September 2-6, Hyderabad, India, 2018, pp.2037–2041.

Shankar R., C. Vikram, and S. Prasanna (2018) “Spoken keyword detection using joint DTW-CNN,” in Proceedings of INTERSPEECH – 19th Annual Conference of the International Speech Communication Associa-tion, September 2-6, Hyderabad, India, pp. 117–121.

Sankar R, Jain A, Deepak KT, Vikram CM, Prasanna SRM (2016) “Spoken term detection from continuous speech using ANN posteriors and image processing techniques”. In: IEEE 22nd national conference on communication (NCC), pp 1–6

Sharma E., G. Ye, W. Wei, R. Zhao, Y. Tian, J. Wu, L. He, E. Lin, and Y. Gong (2020) “Adaptation of RNN transducer with text-to-speech technology for keyword spotting,” in Proceedings of ICASSP – 45th IEEE International Conference on Acoustics, Speech and Signal Processing, May 4-8, Barcelona, Spain, pp. 7484–7488.

Schwarz P, Matejka P, Burget L, Glembek O (2003) “Phoneme recognizer based on long temporal context”. Speech Processing Group, Faculty of Information Technology, Brno University of Technology [Online]. Available: http://speech.fit.vutbr.cz/en/software

Singhal A. (2001) “Modern information retrieval: A brief overview, Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, vol. 24, pp. 35–43.

Sørensen P. M., B. Epp, and T. May (2020) “A depthwise separable convolutional neural network for keyword spotting on an embedded system,” EURASIP Journal on Audio, Speech, and Music Processing, vol. 10, pp.1–14.

Sun M., A. Raju, G. Tucker, S. Panchapagesan, G. Fu, A. Mandal, S. Matsoukas, N. Strom, and S. Vitaladevuni (2016) “Max-pooling loss training of long short-term memory networks for small-footprint keyword spotting,” in Proceedings of SLT – IEEE Spoken Language Technology Workshop, December 13-16, San Diego, USA, pp. 474–480.

Sundar H., J. F. Lehman, and R. Singh (2015) “Keyword spotting in multi-player voice driven games for children,” in Proceedings of INTERSPEECH – 16th Annual Conference of the International Speech Communication Association, September 6-10, Dresden, Germany, pp. 1660–1664.

Sutskever I., O. Vinyals, and Q. Le (2014) “Sequence to sequence learning with neural networks,” in Proceedings of NIPS – 28th International Conference on Neural Information Processing Systems, December 8-13, Montreal, Canada, pp. 3104–3112.

Szöke I (2010) “Hybrid word-subword spoken term detection”. Faculty of Information Technology, BUT, Brno

Szöke I, Schwarz P, Matejka P, Burget L, Karafiát M, Fapso M, Cernocký J (2005b) “Comparison of keyword spotting approaches for informal continuous speech”. In: Interspeech, Citeseer, pp 633–636

Tabibian, S(2020) “A survey on structured discriminative spoken keyword spotting”. ArtifIntell Rev 53, 2483–2520 . https://doi.org/10.1007/s10462-019-09739-y

Tabibian S, Akbari A, Nasersharif B (2013) “Keyword spotting using an evolutionary-based classifier and discriminative features”. Eng Appl ArtifIntell 26:1660–1670

Tabibian S, Akbari A, Nasersharif B (2015) “Speech enhancement using a wavelet thresholding method based on symmetric Kullback–Leibler divergence”. Sig Process 106:184–197

Tabibian S, Shokri A, Akbari A, Nasersharif B (2011) “Performance evaluation for an HMM-based keyword spotter and a large-margin based one in noisy environments”. Proc Comput Sci 3:1018–1022

Tan Y. , K. Zheng, and L. Lei (2019) “An in-vehicle keyword spotting systemwith multi-source fusion for vehicle applications,” in Proceedings of WCNC – IEEE Wireless Communications and Networking Conference, April 15-18, Marrakesh, Morocco.

Tan Z.-H., A. kr. Sarkar, and N. Dehak (2020) “rVAD: An unsupervised segment-based robust voice activity detection method,” Computer Speech & Language, vol. 59, pp. 1–21.

Tang R. and J. Lin (2018) “Deep residual learning for small-footprint keyword spotting,” in Proceedings of ICASSP 2018 – 43rd IEEE International Conference on Acoustics, Speech and Signal Processing, April 15-20, Calgary, Canada, 2018, pp. 5484–5488.

Tang R., W. Wang, Z. Tu, and J. Lin,(2018) “An experimental analysis of the power consumption of convolutional neural networks for keyword spotting,” in Proceedings of ICASSP – 43rd IEEE International Conference on Acoustics, Speech and Signal Processing, April 15-20, Calgary, Canada, pp. 5479–5483.

Thambiratnam AJ (2005) “Acoustic keyword spotting in speech with applications to data mining”. Queensland University of Technology, Brisbane

Thambiratnam K, Sridharan S (2007) “Rapid yet accurate speech indexing using dynamic match lattice spotting”. IEEE Trans Audio Speech Lang Process 15(1):346–357

Tian Y., H. Yao, M. Cai, Y. Liu, and Z. Ma (2021) “Improving RNN transducer modeling for small-footprint keyword spotting,” in Proceedings of ICASSP – 46th IEEE International Conference on Acoustics, Speech and Signal Processing, June 6-11, Toronto, Canada, pp. 5624–5628.

Tucker G., M. Wu, M. Sun, S. Panchapagesan, G. Fu, and S. Vitaladevuni (2016) “Model compression applied to small-footprint keyword spotting,” in Proceedings of INTERSPEECH – 17th Annual Conference of the International Speech Communication Association, September 8-12, San Francisco, USA, pp. 1878–1882.

Vaseghi SV (2008) “Advanced digital signal processing and noise reduction”. Wiley, London

Vasudev D, Gangashetty SV, Anish Babu KK, Riyas KS, (2015) “Query-by-example spoken term detection using Bessel features”. In: IEEE international conference on signal processing, informatics, communication and energy systems (SPICES’15), pp 1–4.

Vaswani A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez,Łukasz Kaiser, and I. Polosukhin (2017) “Attention is all you need,” in Proceedings of NIPS – 31st International Conference on Neural Information Processing Systems, December 4-9, Long Beach, USA, 2017, pp. 5998–6008.

Viikki O, Bye D, Laurila K (1998) “A recursive feature vector normalization approach for robust speech recognition in noise”. In: International conference on acoustics, speech and signal processing (ICASSP’98), pp 733–736

Wallance R, Vogt R, Sridharan S (2007) “A phonetic search approach to the 2006 NIST spoken term detection evaluation”. In: INTERSPEECH, pp 2385–2388

Wallance R, Vogt R, Sridharan S (2009) “Spoken term detection using fast phonetic decoding”. In: Proceedings of ICASSP, pp 4881–488474

Wang D. (2010) Out-of-vocabulary spoken term detection. Université de Edinburgh,

Edinburgh

Wang D, Tejedor J, Frankel J, King S, Colás J (2009) “Posterior-based confidence measures for spoken term detection”. In: International conference on acoustics, speech and signal processing(ICASSP’09), pp 4889–4892

Wang D, Tejedor J, King S, Frankel J (2012) “Term-dependent confidence normalisation for out-of-vocabulary spoken term detection”. J Comput Sci Technol 27:358–375

Wang H, Lee T, Leung C (2011) “Unsupervised spoken term detection with acoustic segment model”. In: International conference on speech database and assessments (Oriental COCOSDA), pp 106–111

Wang L., R. Gu, N. Chen, and Y. Zou (2021) “Text anchor based metric learning for small-footprint keyword spotting,” in Proceedings of INTERSPEECH – 22nd Annual Conference of the International Speech Communication Association, August 30-September 3, Brno, Czechia, pp. 4219–4223.

Wang X., S. Sun, and L. Xie (2019a) “Virtual adversarial training for DS-CNN based small-footprint keyword spotting,” in Proceedings of ASRU – IEEE Automatic Speech Recognition and Understanding Workshop, December 14-18, Singapore, Singapore pp. 607–612.

Wang X., S. Sun, C. Shan, J. Hou, L. Xie, S. Li, and X. Lei (2019b) “Adversarial examples for improving end-to-end attention-based small-footprint keyword spotting,” in Proceedings of ICASSP – 44th IEEE International Conference on Acoustics, Speech and Signal Processing, May 12-17, Brighton, UK, pp. 6366–6370.

Wang Y. and Y. Long (2018) “Keyword spotting based on CTC and RNN for Mandarin Chinese speech,” in Proceedings of ISCSLP – 11th International Symposium on Chinese Spoken Language Processing, November 26-29, Taipei, Taiwan, pp. 374–378.

Warden. P (2017) Launching the Speech Commands Dataset. [Online]. Available: https://ai.googleblog.com/2017/08/launching-speech-commands-dataset.html

Warden.P (2018) “Speech Commands: A dataset for limited-vocabulary speech recognition,” arXiv:1804.03209v1.

Watanabe S., M. Delcroix, F. Metze, and J. R. Hershey (2017) “New Era for Robust Speech Recognition” . Springer.

Wei. B, M. Yang, T. Zhang, X. Tang, X. Huang, K. Kim, J. Lee, K. Cho, and S.-U. Park (2021) “End-to-end transformer-based open-vocabulary keyword spotting with location-guided local attention,” in Proceedings of INTERSPEECH – 22nd Annual Conference of the International Speech Communication Association, August 30-September 3, Brno, Czechia, pp. 361–365.

Weintraub M (1995) “LVCSR log-likelihood ratio scoring for keyword spotting.” In: International conference on acoustics, speech, and signal processing (ICASSP-95), pp 297–300

Wilpon, J. G., Lee, C. H., Rabiner, L. R. (1989). Application of hidden Markov models for recognition of a limited set of words in unconstrained speech. In International Conference on Acoustics, Speech, and Signal Processing, (pp. 254-257).

Wöllmer M., B. Schuller, and G. Rigoll (2013) “Keyword spotting exploiting long short-term memory,” Speech Communication, vol. 55, pp. 252–265.

Wu H., Y. Jia, Y. Nie, and M. Li (2020) “Domain aware training for far-field small-footprint keyword spotting,” in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication Association, October 25-29, Shanghai, China, pp. 2562–2566.

Xu M. and X.-L. Zhang (2020) “Depthwise separable convolutional ResNet with squeeze-and-excitation blocks for small-footprint keyword spotting,” in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication Association, October 25-29, Shanghai, China, pp. 2547–2551.

Xuan X., M. Wang, X. Zhang, and F. Sun (2019) “Robust small-footprint keyword spotting using sequence-to-sequence model with connectionist temporal classifier,” in Proceedings of ICICSP – 2nd International Conference on Information Communication

and Signal Processing, September 28-30, Weihai, China, pp. 400–404.

Yan H., Q. He, and W. Xie (2020) "CRNN-CTC based Mandarin keywords spotting " in Proceedings of ICASSP – 45th IEEE International Conference on Acoustics, Speech and Signal Processing, May 4-8, Barcelona, Spain, pp. 7489–7493.

Yang C., X. Wen, and L. Song (2020) "Multi-scale convolution for robust keyword spotting," in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication Association, October 25-29, Shanghai, China, pp. 2577–2581.

Yılmaz E., Özgür Bora Gevrek, J. Wu, Y. Chen, X. Meng, and H. Li (2020) "Deep convolutional spiking neural networks for keyword spotting," in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication Association, October 25-29, Shanghai, China, pp. 2557–2561.

Yoshizawa S, Hayasaka N, Wada N, Miyanaga Y (2004) "Cepstral gain normalization for noise robust speech recognition". In: International conference on acoustics, speech, and signal processing (ICASSP'04), pp I-209–I-212

Yu M., X. Ji, B. Wu, D. Su, and D. Yu (2020) "End-to-end multi-look keyword spotting," in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication Association, October 25-29, Shanghai, China, pp. 66–70.

Yuan Y., Z. Lv, S. Huang, and L. Xie (2019) "Verifying deep keyword spotting detection with acoustic word embeddings," in Proceedings of ASRU – IEEE Automatic Speech Recognition and Understanding Workshop, December 14-18, Singapore, Singapore, pp. 613–620.

Zeng M. and N. Xiao (2019) "Effective combination of DenseNet and BiLSTM for keyword spotting," IEEE Access, vol. 7, pp. 10 767–10 775.

Zeppenfeld, T., Waibel, A. H. (1992). A hybrid neural network, dynamic programming word spotter. In Acoustics, Speech, and Signal Processing, IEEE

International Conference on (Vol. 2, pp. 77-80).

Zeppenfeld, T., Houghton, R., Waibel, A. (1993). Improving the MS-TDNN for word spotting. In 1993 IEEE International Conference on Acoustics, Speech, and Signal Processing (Vol. 2, pp. 475-478).

Zhang B. , W. Li, Q. Li, W. Zhuang, X. Chu, and Y. Wang (2021) "AutoKWS:Keyword spotting with differentiable architecture search," in Proceedings of ICASSP – 46th IEEE International Conference on Acoustics, Speech and Signal Processing, June 6-11, Toronto, Canada, pp.2830–2834.

Zhang Y, Glass JR (2009) Unsupervised spoken keyword spotting via segmental DTW on Gaussian posteriorgrams. In: IEEE workshop on automatic speech recognition & understanding, IEEE, pp 398–403.

Zhang P. and X. Zhang (2020) "Deep template matching for small-footprint and configurable keyword spotting," in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication Association, October 25-29, Shanghai, China, pp. 2572–2576.

Zhang Y., N. Suda, L. Lai, and V. Chandra (2018) "Hello edge: Keyword spotting on microcontrollers," arXiv:1711.07128v3.

Zhao Z. and W.-Q. Zhang (2020) "End-to-end keyword search based on attention and energy scorer for low resource languages," in Proceedings of INTERSPEECH – 21st Annual Conference of the International Speech Communication Association, October 25-29, Shanghai, China, pp. 2587–2591.

Zhuang Y., X. Chang, Y. Qian, and K. Yu (2016) "Unrestricted vocabulary keyword spotting using LSTM-CTC," in Proceedings of INTERSPEECH – 17th Annual Conference of the International Speech Communication Association, September 8-12, San Francisco, USA, 2016, pp. 938– 942.