Story Generation from Smart Phone Data : A script approach

Trung Ky Nguyen

Communauté
UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTE UNIVERSITE GRENOBLE ALPES

Spécialité : **Informatique**

Arrêté ministériel : 25 mai 2016

Présentée par

Ky Trung NGUYEN

Thèse dirigée par **Catherine GARBAY,** co-directeur par **François PORTET**
préparée au sein du **Laboratoire d'Informatique de Grenoble** dans **l'École Doctorale Ingénierie pour la Santé la Cognition et l'Environnement**

Story Generation from Smartphone Data: A Script Approach

Thèse soutenue publiquement le **16 décembre 2019**,
devant le jury composé de :

M. Dan ISTRATE
Enseignant-Chercheur, HDR, UTC, Compiègne, rapporteur
M. Anthony FLEURY
Maître de Conférences, HDR, IMT Lille Douai, rapporteur
M. Norbert NOURY
Professeur, Université Claude Bernard Lyon, examinateur
Mme. Paule-annick DAVOINE
Professeur, Université de Grenoble-Alpes, président
Mme. Catherine GARBAY
Directrice de recherche CNRS, HDR, LIG, directrice de thèse
M. François PORTET
Maître de Conférences, Grenoble-INP, LIG, co-directeur de thèse

Invitée:
Mme. Sylvie CHARBONNIER
Maitre de conférences, HDR, Université Grenoble Alpes

Résumé

Le script est une structure qui décrit une séquence stéréotypée d'événements ou d'actions survenant dans notre vie quotidienne. Les histoires utilisent des scripts, avec une ou plusieurs déviations intéressantes, qui nous permettent de mieux saisir les situations quotidiennes rapportées et les faits saillants du récit. Ainsi, la notion de script est très utile dans de nombreuses applications d'intelligence ambiante telles que la surveillance de la santé et les services d'urgence. Ces dernières années, l'avancement des technologies de détection et des systèmes intégrés permettent aux systèmes de santé de collecter en permanence les activités des êtres humains, en intégrant des capteurs dans des dispositifs portables (par exemple smart-phone ou smart-watch). La reconnaissance de l'activité humaine (HAR) a ainsi connue un essor important grâce notamment à des approches d'apprentissage automatique telles que le réseau neuronal ou le réseau bayésien. Ces avancées ouvre des perspectives qui vont au delà de la simple reconnaissance d'activités. Ce manuscrit défend la thèse selon laquelle ces données de capteurs portables peuvent être utilisées pour générer des récits articulés autour de scripts en utilisant l'apprentissage automatique. Il ne s'agit pas d'une tâche triviale en raison du grand écart sémantique entre les informations brutes de capteurs et les abstractions de haut niveau présente dans les récits. A notre connaissance, il n'existe toujours pas d'approche pour générer un une histoire à partir de données de capteurs en utilisant l'apprentissage automatique, même si de nombreuses approches d'apprentissage automatique (réseaux de neurones convolutifs, réseaux de neurones profonds) ont été proposées pour la reconnaissance de l'activité humaine au cours des dernières années. Afin d'atteindre notre objectif, nous proposons premièrement dans cette thèse un nouveau cadre qui traite le problème des données non uniformément distribuées (problème du biais induit par des classes majoritaires par rapport aux classes minoritaires) basé sur un apprentissage actif associé à une technique de sur-échantillonnage afin d'améliorer la macro-exactitude de classification des modèles d'apprentissage classiques comme le perception multi-couche. Deuxièmement, nous présentons un nouveau système permettant de générer automatiquement des scripts à partir de données d'activité humaine à l'aide de l'apprentissage profond. Enfin, nous proposons une approche pour l'apprentissage de scripts à partir de textes en langage naturel capable d'exploiter l'information syntaxique et sémantique sur le contexte textuel des événements. Cette approche permet l'apprentissage de l'ordonnancement d'événements à partir d'histoires décrivant des situations typiques de vie quotidienne. Les performances des méthodes proposées sont systématiquement discutées sur une base expérimentale.

Résumé du Chapter 1

Ce chapitre introduit la thèse par une présentation des motivations qui nous ont conduit à aborder la génération d'histoires à partir de données d'activité issues d'un outil de type smartphone. Il en discute les principales difficultés, et donne un aperçu de notre recherche et de nos principales contributions pour les aborder. En fait, de nos jours, les données des capteurs portables sont collectées partout. Cependant, ces données sont souvent représentées sous une forme binaire et numérique, ce qui est difficile à comprendre pour l'homme, et il n'existe toujours pas d'approche

permettant de les structurer et de les transformer sous une forme clairement accessible (par exemple, une histoire ou des scripts). Nous insistons aussi dans ce chapitre sur le rôle joué par l'apprentissage automatique dans ce processus qui met en jeu une chaîne complexe de traitement, et sur les limites des techniques conventionnelles, centrées sur une problématique d'abstraction des données. Par l'apprentissage profond, nous centrons notre attention sur les corrélations entre informations et éléments cachés de contexte, au sein et entre niveaux d'abstraction. Trois enjeux majeurs sont finalement identifiés: prise en compte du déséquilibre des données pour la reconnaissance d'événements, modèles d'apprentissage profonds pour la génération flexible de concepts sémantiques et de scripts à partir d'événements, apprentissage de scripts à partir de textes en langage naturel capables d'exploiter l'information syntaxique et sémantique sur le contexte textuel des événements. Enfin, la dernière section présente un aperçu des différents chapitres de la thèse.

Résumé du Chapter 2

Ce chapitre présente l'état de l'art en le structurant en deux grandes sections : (i) transformation de données en textes et génération de scripts à partir de données, (ii) reconnaissance de l'activité humaine à partir de données ambiantes. Dans la première section, nous présentons tout d'abord quelques grandes tendances de la transformation de données en texte. Dans ces méthodes, centrées sur la génération automatique de textes en langue naturelle (NLG), l'accent est peu mis sur la problématique de l'apprentissage automatique. Nous discutons ensuite la problématique de la génération de scripts utilisant le traitement de langage naturel (NLP) et les méthodes conventionnelles d'apprentissage automatique mises en jeu. Les aspects liés à la modélisation des événements et de leur structure temporelle sont particulièrement mis en avant. On notera néanmoins que ces travaux s'appliquent non pas à des données ambiantes mais à des textes en langue naturelle. La section suivante du chapitre présente l'état de l'art de la reconnaissance de l'activité humaine (HAR) à partir de données de capteurs portables en utilisant l'apprentissage automatique. Premièrement, nous introduisons la problématique de la reconnaissance de l'activité humaine. Nous présentons ensuite les méthodes pour réaliser la reconnaissance de l'activité, en insistant d'abord sur les questions de l'extraction de caractéristiques et des métriques pour l'évaluation de ces méthodes. Nous consacrons ensuite deux sections à la problématique de l'apprentissage automatique, la première dédiée aux techniques conventionnelles, la seconde aux techniques de l'apprentissage profond.

Résumé du Chapter 3

Ce court chapitre donne un aperçu de la problématique abordée dans la thèse, de ses dimensions, et de la démarche proposée. La problématique est présentée comme associant une dimension d'abstraction des données et une dimension cognitive de communication. L'abstraction des données met en jeu la qualité des données (introduction de la notion de déséquilibre), la prise en compte du contexte (what, when and where) mais aussi celle de la traduction entre univers de discours (du numérique au langagier) différents. La communication des données d'activité met en jeu la notion de script véhiculant une information temporelle. Elle implique la prise en compte d'une dimension linguistique. Des questions difficiles d'apprentissage sont impliquées dans les 2 dimensions et nous mettons en avant l'apport des techniques récentes de l'apprentissage profond. Nous présentons la

démarche proposée dans la dernière section. Elle prend appui sur les 2 domaines de l'apprentissage automatique et du traitement du langage naturel.

Résumé du Chapter 4

Ce chapitre présente le cadre de notre proposition pour traiter les données déséquilibrées provenant des données de capteurs de téléphones mobiles. Nous donnons une introduction sur la question du déséquilibre dans la première section, puis nous présentons les travaux connexes qui traitent de ce problème. Ensuite, dans la section 4.3 nous présentons notre nouveau cadre, qui utilise une technique d'apprentissage actif et de suréchantillonnage pour résoudre les problèmes de données de déséquilibre, basé sur le perceptron Multicouche. Ensuite, les résultats de l'expérience sont illustrés dans la section 4.4 sur 2 jeux de données comportant des données audio et d'accéléromètre, toutes 2 comportant des données déséquilibrées. L'apport de notre méthode est clairement démontré. Le chapitre se termine par une conclusion.

Résumé du Chapter 5

Dans ce chapitre, nous présentons notre nouvelle approche pour générer des scripts à partir d'événements, qui sont détectés à l'aide d'une méthode d'apprentissage en profondeur sur des données de capteurs portables. Après une introduction en section 5.1, les travaux associés sont présentés dans la section 5.2. La méthode proposée est détaillée dans la section 5.3. Elle comporte 2 étapes: génération de concepts sémantiques à partir des données capteur, génération de scripts à partir des concepts. Les concepts sémantiques sont définis comme des triplets associant geste, activité, contexte. Ils sont générés via des réseaux de neurones profonds et ordonnés temporellement. La génération de scripts s'effectue via une méthode hybride associant une phase de détection à base de patrons et une phase de traduction de ce "langage source" sous la forme de phrases en langage cible (scripts).

Les résultats expérimentaux sont présentés dans la section 5.4, sur un ensemble de données d'activités journalières impliquant plusieurs sujets. Les performances des 2 phases de traitement sont évaluées séparément. Le chapitre se termine par une brève discussion et un aperçu des travaux futurs dans la section 5.5.

Résumé du Chapter 6

Nous présentons dans ce chapitre, une approche pour apprendre des scripts à partir de textes en langage naturel. Dans la section 6.1, nous présentons une introduction sur la notion de script et son intérêt pour la reconnaissance d'activités humaines. Nous proposons un bref panorama des travaux dans le domaine de l'apprentissage de script et donnons les bases de notre contribution. La section 6.2 est dédiée à la présentation des travaux connexes: modèles de représentation des événements, en particulier multi-arguments et compositionnels, modélisation et apprentissage des séquences d'événements. Dans la section 6.3, nous détaillons les méthodes proposées et présentons notre modèle NESS, fondé sur l'exploitation des réseaux de neurones récursifs et une représentation des événements comme composition de leurs arguments sémantiques. Les résultats expérimentaux sont présentés dans la section 6.4. Nous comparons les performances de notre approche avec celles d'autres méthodes de la littérature sur une base de données publique et montrons son apport avant de conclure en section 6.5.

Résumé du Chapter 7

Ce chapitre est dédié à la présentation de nos conclusions et perspectives. Il rassemble nos principales propositions dans les différents domaines scientifiques que nous avons couvert: reconnaissance de l'activité humaine, génération et apprentissage de scripts via les traitements du langage naturel. Nous revenons sur la variété et la nouveauté des outils et modèles utilisés et insistons sur leur complémentarité pour faire face aux difficultés (i) du monde numérique des données capteur et (ii) du monde sémantique de la représentation de l'activité en langue naturelle. Nous revenons sur le rôle pivot de la notion de script et sur l'apport de l'apprentissage profond pour affronter ces difficultés. Nous abordons ensuite les limitations de ce travail: pauvreté des données, caractère répétitifs des événements, manque de représentativité des activités, manque de richesse des ressources langagières. Les principales perspectives concernent l'introduction d'autres données, comme l'émotion ou la physiologie. Enfin, nous rappelons l'apport de cette thèse à différentes applications du monde réel.

Abstract

Script is a structure describes an appropriate sequence of events or actions in our daily life. A story has invoked a script with one or more interesting deviations, which allows us to deeper understand what was happened in the routine behavior of our daily life. Therefore, it is essential in many ambient intelligence applications such as health-monitoring and emergency services. Fortunately, in recent years, with the advancement of sensing technologies and embedded systems, which make the health-care system possible to collect activities of human beings continuously, by integrating sensors into wearable devices (e.g., smart-phone, smart-watch, etc.). Hence, human activity recognition (HAR) has become a hot topic interest in research over the past decades. In order to do HAR, most researches used machine learning approaches such as Neural networks, Bayesian networks, etc. Therefore, the ultimate goal of our thesis is to generate such kind of stories or scripts from activity data of wearable sensors using a machine learning approach.

However, to the best of our knowledge, it is not a trivial task due to the very limitation of information on wearable sensors activity data. Hence, there is still no approach to generate script/story using machine learning, even though many machine learning approaches were proposed for HAR in recent years (e.g., convolutional neural network, deep neural network, etc.) to enhance the activity recognition accuracy.

In order to achieve our goal, first of all in this thesis we proposed a novel framework, which solved for the problem of imbalanced data, based on active learning combined with oversampling technique so as to enhance the recognition accuracy of conventional machine learning models i.e., Multilayer Perceptron. Secondly, we introduce a novel scheme to automatically generate scripts from wearable sensor human activity data using deep learning models, evaluate the generated method performance. Finally, we proposed a neural event embedding approach that is able to benefit from semantic and syntactic information about the textual context of events. The approach is able to learn the stereotypical order of events from sets of narrative describing typical situations of everyday life.

Acknowledgments

I would like to express my deepest gratitude to my thesis supervisors, Dr. Catherine GARBAY and Dr. Francois PORTET, for providing me with extensive support, patience throughout the duration of my Ph.D. This thesis would be impossible without their guidance and support during difficult times.

I would like to thank my core jury members, Dr. Dan ISTRATE, Dr. Anthony FLEURY, Professor Norbert NOURY, Professor Paule-annick D'AVOINE, and Dr. Sylvie CHARBONNIER for providing the most constructive and invaluable comments and suggestions.

I would like to express my sincere gratefulness to my colleagues (Dr. Raheel Qader, Dr. Nguyen Van Bao, To Thanh Hai, Tran Thi Phuong Thao, Nguyen Thi Thanh Quynh) who have been generous with their help, inspiration, and encouragement. Especially, I would like to thank Dr. Mai Thai Son for all the "what"s, "why"s and "how"s and for all the collaboration we could achieve.

I would like to express my sincere thanks to the Ministry of Education and Training of Vietnam in partnership with the French government (Campus France), who awarded me a doctoral scholarship, and with which I devoted myself to the establishment of the thesis. I also would like to thank my supervisors for their role in partial funding support for me throughout the duration of my Ph.D.

I would like to thank my family for making this happen with their unconditional love and support all the time of my life. Especially, I would like to take the opportunity for an offcial acknowledgment for my wife (DANG Thi Hong Hue) and our children for their support and motivation during the whole of my Ph.D.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

1.1 Story Generation at the Age of Big Data

If the rise of big data has opened many opportunities, making sense of this mass of information has become one of the major challenges of this century. In this thesis, we are focusing on an original approach to deal with a subset of big data, namely personal ambient data (e.g., smartphones, GPS, wearable sensors, etc.), by presenting the collected data in the form of a personal story. Indeed, human has developed cognitive capabilities specifically tuned to tell and understand the information in the form of stories [Reiter and Dale (2000); Portet et al. (2009); Hunter et al. (2012)].

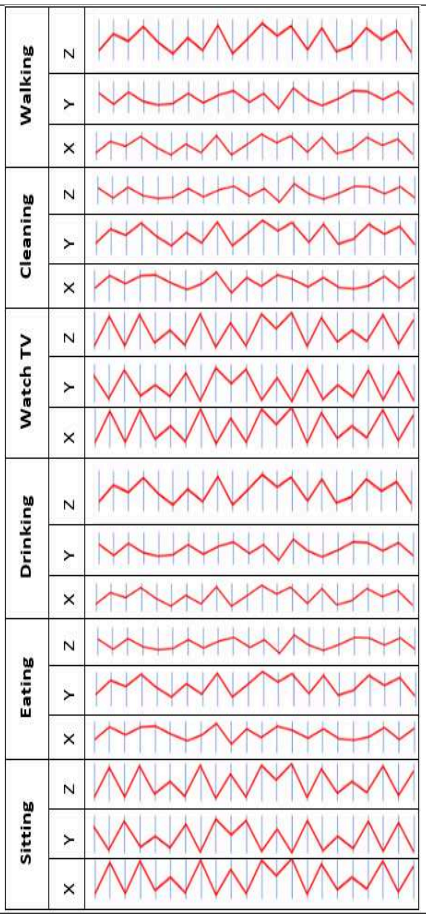
Automatic story generation (SG) has been studied in the domain of Natural Language Generation (NLG). NLG is concerned with the automatic generation of written texts from temporal data to support various domains of daily life such as: the generation of weather report from weather simulations [Reiter and Dale (2000); Reiter et al. (2005)]; helping the patients understand their complex medical data [Portet et al. (2009); Hunter et al. (2012)]; or supporting social communication for disable people Williams and Reiter (2008). Many story generators have been proposed [Andersen et al. (1992); Jacobs and Rau (1990); Poibeau et al. (2013)]. However, most of these approaches were not interested in generating a story from real data but rather from virtual environments where all the information is complete and known. Although NLG has a long history of generative narratives from data or knowledge, most of these narratives do not concern personal ambient data. In Ambient Intelligence, data is captured by the sensor and thus filled with uncertainty, incomplete and very low level (e.g., the measure of the skin temperature but not whether the person feels too warm or cold). Therefore, our ultimate goal is to generate automatically stories from ambient data, and the Ph.D. subject is focused on **Story Generation from Smart-phone Data**. As will be described in this thesis, story generation from smart-phone data concern the definition of a method to extract meaning information from real sensor

Figure 1.1: Relationship between data and their representation modality.

data, to organize them in the form of a story and to express this story through a narrative. This method would allow humans to get an insight into the real data. Figure 1.1 illustrates the relationship between three key elements: data, visual and narrative. In fact, the narrative helps to explain what's happening in the data with ample context and commentary, while a visual representation of data mostly enlightens some aspect of the data without necessarily providing the context to understand them Kashimoto et al. (2017). Moreover, when narrative and visuals are merged together, they can engage or entertain humans.

Ubiquitous computing (e.g., mobile computing, ambient intelligence, etc.) is a rich domain that enables us to implement and evaluate real use cases from users who can share their stories through online portals. Such stories include environment data (e.g., weather, altitude, human activity, etc.), temporal data, affective/evaluative data. Figure 1.2 shows an example of a story written by a user (right side) and the ambient data that was collected (left side). This figure also illustrates our ultimate purpose, which aims to find technology in order to collect real data from sensor devices (e.g., accelerometers, gyroscope, etc.) as represented in the left side of the figure, and then translate these raw signal into a story as presented in the right side of the figure. Since this story is readable by a human it is easier to understand what the data contains than looking at the raw signals. The story describes about what happened after the girl woke up and describes low-level activities (e.g., sitting, walking, etc.) as well as high-level activities (e.g., drink coffee, clean up the floor, etc.), after that how she reacted with her mother talking about the supermarket such as emotion (e.g., exciting, sad, etc.) and feeling (e.g., warm, cold, etc.), then how they decided to go shopping.

This example illustrates some of the challenges that have to be addressed if we want to generate automatically a story at a human level only from sensors data. The information must be extracted from raw signals, information must be structured and linked to the



At 10:00, she sat on a sofa in the living room and took a coffee with crackers because she was not fully awake.
After that, she turned on television and watched a musical game-show, while her mother was cleaning up the floor.
At 10:30 her mother told her to go to supermarket. Then, she seemed very exciting with it, and went to prepare her dress quickly.
At 10:50 They left home and went shopping in the supermarket.

Figure 1.2: Example of story written by a user (right side) and the smart-phone data was collected during an experiment (left side).

situation in which the data is collected. The structured information must be organized to respect the conventional human-understandable stories and such a story must be expressed via a textual representation.

Being able to solve such problems involve many challenges. First, each of the above steps are related to different research domains (data analysis, knowledge reasoning, Natural Language Generation) and are generally addressed as a pipeline of processing steps (reprocessing of data, segmentation, event extraction, script recognition, textual generation). In the state of the art, such processing is performed using a model acquired through machine learning, however, this application suffers from the sparsity of information. Indeed, if it is possible to find data-set related to smartphones and activity, it is extremely hard to find a data-set containing both annotated smartphone data and their representation textual stories.

Second, there is a wide semantic gap between the raw signal and the final story. To deal with this gap, each step has to consider the overall objective of the task of story generation. This is very difficult to perform in a pipeline of successive steps. Furthermore, with such a chain of abstraction steps, each optimized separately, hidden relationships between information across the abstraction levels that may not be captured. Current methods still rely on a pipeline approach and lack to consider the correlation between information inside and across the various abstraction levels. Going further in these directions raises still unsolved issues in modeling and learning. We explore some of them in this thesis, based on the recent deep learning approaches.

In this thesis, we present the work performed to move toward story generation from sensor data by using deep learning neural networks. In particular, the thesis work was focused on Human Activity Recognition (HAR) from sensors, Concept-to-Text generation from abstract events, and scripts learning to extract a typical sequence of events (e.g., sub-stories) that can be used to model the structure of a story. Hence, this approach is seen as a bridge that would allow connecting between the domain of HAR, scripts modeling and NLG in the NLP domain.

1.2 Research Overview and Contributions

In this thesis, **Human Activity Recognition** (HAR) is the first target of the research orientation. In HAR, several issues were addressed in [He and Garcia (2009); Lara et al. (2012)] such as imbalanced data, flexibility, etc. Second, the HAR system only allows one to recognize discrete human daily activities, hence it is not sufficient to build or generate a story as described in Miyanishi et al. (2018), which is defined as a problem in NLG domain. Finally, we would like to predict or infer missing events on scripts modeling from

the narrative text, which is a hot topic in NLP [Modi et al. (2016); Frermann et al. (2014); Regneri et al. (2010)]. Therefore, considering all of these problems in different domains, we summarize the contributions of the thesis is the following:

- *Dealing with Imbalanced Datasets for Human Activity Recognition using Mobile Phone Sensors*: In the recent years, the wide spreading of smart-phones that are daily carried by humans and fit with tens of sensors triggered an intense research activity in human activity recognition (HAR). HAR in smart-phone is seen as essential not only to better understand human behavior in daily life but also for context provision to other applications in smart-phones. Many statistical and logical based models for on-line or off-line HAR have been designed, however, the current trend is to use deep-learning with neural networks. These models need a high amount of data and, as most discriminative models, they are very sensitive to the imbalanced class problem He and Garcia (2009). In this work, we study different ways to deal with imbalanced data sets to improve the accuracy of HAR with neural networks and introduce a new over-sampling method, called Border Limited Link SMOTE (BLL SMOTE) that improves the classification accuracy of Multi-Layer Perceptron (MLP) performances, which is presented in Chapter 4.

- *Automatic Scripts Generation from Human Activity Recognition using Wearable Sensors Data*: Human activity recognition (HAR) plays an important role in the real world, which allows us to recognize activities of daily living (ADLs) and understand human life. In spite of this task being an active field in the past decade, most existing researches concentrate on predicting individual activity labels from internal and external sensors data. However, our daily life scripts are composed of chronological ADLs with the corresponding locations and higher activities (e.g., shopping in the supermarket after eating at the kitchen and leaving home). Therefore, in this work, we proposed a novel approach that generates scripts from human activity recognition using wearable sensors data. The proposed method combines semantic concepts such as gesture, activity, and location detected by sensors for producing a script with the above three real-world properties. First, we used deep learning neural networks (DNN) to recognize concepts in terms of a sequence (e.g., gesture, activity, and location) from wearable sensors data. Second, we combine these concepts by using manually detection templates in order to glue the concepts in terms of a sentence. Finally, we formulated the natural language generation step as a machine translation approach, by applying the sequence to sequence model Sutskever et al. (2014a) to enrich the target generated sentence from their semantic concepts, which is considered as the source language. This work is presented in Chapter 5.

- *Neural Networks Modeling for Semantic Script Induction from Narrative Text*: Typical situations of everyday life such as going to a restaurant are pre-encoded common sense knowledge known as *Scripts*. Learning such scripts from narratives has been studied in Natural Language Processing (NLP). These script models would allow predicting likely next events and would support the natural generation of narratives. Early work of *Script learning* focused on word frequency-based methods that suffered from the sparsity problem. Therefore, event embeddings has raised interest to overcome this issue [Modi (2016); Modi and Titov (2014)]. In this work, we proposed a neural event embeddings approach that is able to benefit from semantic and syntactic information about the textual context of events. The approach is able to learn the stereotypical order of events from sets of narratives describing typical situations of everyday life. This work is described in Chapter 6.

1.3 Thesis Outline

Chapter 2 gives a short overview of the state-of-the-art on both data-to-text generation and script learning and generation. Then, it presents human activity recognition (HAR) from mobile phone sensors, which focuses on the definition of the problem, and the way this task is generally evaluated. It also introduces a review of state-of-the-art on human activity recognition, which describes different methods and works, using for HAR on both machine learning approaches such as conventional and deep learning.

Chapter 3 describes the scientific challenges, the overall approach of the thesis and the design of the framework to generate stories from raw smartphone sensor data.

In Chapter 4 we introduce our novel framework, based on Multilayer Perceptron (MLP), which uses active learning and oversampling technique so that it can solve the problems of imbalanced data.

In Chapter 5, we proposed a novel approach allows us to generate a story in terms of *Scripts* from human activity recognition using wearable sensor data and evaluation for *Scripts generation*.

Chapter 6 presents a novel technique to classify the order of events from the sequence of events or *scripts* from natural language texts. This technique permits to predict what event happens next from learning ordered of the sequence of events.

Chapter 7 provides an overall conclusion about the work performed during the thesis and discusses the numerous perspectives for further research it opens.

Chapter 2

State-of-the-art in Human Activity Recognition and Story Generation

Scenario 1: Preparing clothes
Find dirty clothes
Put dirty clothes in the basket
Take clothes to washing machine
Scenario 2: Washing clothes
Put clothes in washing machine
Add laundry detergent
Set washing machine to desired setting
Turn on washing machine
Scenario 3: Drying clothes
When finished put clothes in dryer
When dryer finished fold clothes
Put clothes back to the basket
Scenario 4: Exiting laundry
Turn off dryer
Leave laundry room

Table 2.1: Event sequence of laundry

Scripts is a structure that describes an appropriate sequence of events in a particular context, was first introduced by Schank and Abelson (1975). The structure is an interconnected whole and what in one slot affects to what can be in another. As described in Schank and Abelson (1975) story is a script with various interesting deviations. Such numerous scripts describe different sequence of events or actions in our daily life such as bus taking script, working day script and laundry script, and so on. For instance, Table

2.1 shows an example of laundry scripts in different scenarios.

The above scripts enable us an insight into the sequence of events happening in the laundry context with different scenarios, which are providing in the form of human-readable texts. For instance, the sequence of events in scenario 2 lets us deduce that *the person is washing clothes using a washing machine*, or scenario 3 shows us that *the person is drying his/her clothes*. In general, the overall script would represent the prototypical sequence of events that human would imagine being part of this task. Hence script is used by humans to interpret scenes (she doing the laundry) predict the next event and do common sense reasoning. Since real-life stories are intended to report human activities in a way that is tuned to the human capability of the understanding script, we introduce the state of the art in reasoning with the script and human activity recognition from a computer science perspective.

2.1 State of the art of Data-to-text and Script Generation

In the several past decades have witnessed a significant researches on data-to-text generation and script generation. As our literature surveys on data-to-text and script generation, we can divide it into two domains as follows:

For data-to-text using Natural Language Generation, Reiter et al. (2005) introduced a SUMTIME-MOUSAM system to automatically generate textual weather forecast by choosing the right word to communicate numeric weather prediction data. The input to SUMTIME-MOUSAM system is numerical weather parameters and the output is the weather forecast. However, the input parameters of SUMTIME-MOUSAM system are produced by simulation. In addition, story generation is not only interested in weather forecast but also studied in health-care such as helping the patients understand their complex medical data in [Portet et al. (2009); Hunter et al. (2012)] presented BT-45 system which generates textual summaries of about 45 minutes of continuous physiological signal (e.g., heart rate, pressure of oxygen and carbon dioxide in blood, oxygen saturation, mean blood pressure, and peripheral and temperature of the baby) and discrete events (e.g., equipment settings and drug administrations). As a result, an experiment on clinical data of BT-45 system shows that generated texts are inferior to human expert texts in a number of ways. Likewise, Williams and Reiter (2008) proposed the SKILLSUM system that generates personalized feedback reports for people with limited reading skills. In order to do this, there are two approaches to the SKILLSUM system which the first approach is to determine the content and structure (document-planning) by using pilot experiments

and interview with domain experts. The second approach is choosing linguistic expressions (micro-planning) by using constraints based on corpus analysis and preference rules. The evaluation of the SKILLSUM system illustrated that the generated text was more effective than canned text at enhancing users' knowledge of their skills. Moreover, many story generators have been proposed [Andersen et al. (1992); Jacobs and Rau (1990); Poibeau et al. (2013)]. Andersen et al. (1992) introduced JASPER which is a fact extraction system. The system used a template-driven approach, partial understanding technique and heuristic procedures to extract certain important pieces of information from a limited range of text. Nevertheless, these approaches were not interested in generating a story from real data but rather from virtual environments where all the information is complete and known. In more recent research, Miranda (2018) works on story generation from real sensor data, which allows generating a sequence of human activities during their ski-touring. In order to generate the sequence of activities, the proposed system used signal segmentation with threshold from altitude, which was collected from the global positioning system (GPS) of the smart-phone to extract information and knowledge about human activities such as moving, standing as well as the interest location. Then, it is mapped into an ontology tool with data-to-text processes (e.g., micro-planning, etc.) to generate the story. However, different from this work, our thesis only focuses on the machine learning approach for recognizing human activities.

For script generation, early works conducted in the 1970s defined a script as a sequence of structured events organized in temporal order Schank and Abelson (1975). Scripts learning was revived by Chambers and Jurafsky (2008) work, which presented a statistical approach to capture script knowledge. They proposed an unsupervised induction framework, called narrative event chains to infer event and rich understanding from raw news-wire text. Narrative event chains were automatically extracted, by following mentions of an entity (e.g., protagonist) through the narrative text, which is detected on the outputs of a coreference resolution system and a dependency parser. The relationship between events was computed using Pairwise Mutual Information (PMI) score. The model's ability to capture commonsense knowledge was evaluated using the Narrative Cloze (NC) task, in which one event is removed from the event chain and the model is evaluated by ranking all candidate events. Likewise, in Regneri et al. (2010) work proposed approach to unsupervised learning scripts, which focused on temporal event structure of scripts by building a *temporal script graph*. The graph was calculated for a scenario by identifying corresponding event descriptions using a Multiple Sequence Alignment (MSA) algorithm, and converting alignment into a graph. At the evaluation phase, the script graph algorithm showed that it significantly outperforms a clustering-based baseline, and the algorithm can distinguish event descriptions that appear at different points in the script

story-line. Another unsupervised learning of scripts knowledge from the natural text was proposed by Frermann et al. (2014), which used a hierarchical Bayesian model by inducing jointly events and constraints on event ordering in one unified framework. They refer to two types of entities in each type of scenarios such as event types and participant types. Then, they incorporated the Generalized Mallows Model (GMM) over orderings from sets of *Event Sequence Descriptions* (EDSs). In the evaluation stage, their system outperforms the MSA algorithm provided by Regneri et al. (2010) on the task of event ordering and achieving comparable results in the event induction task.

In the same year, Pichotta and Mooney (2014) proposed a script learning approach that employs events with multiple arguments. Unlike previous works, they model the interactions between multiple entities that enable to better prediction of events in natural texts or documents. A multi-argument event is a relational atom $a = v(e_s, e_o, e_p)$, where v is a verb lemma, and e_s, e_o, e_p are a subject relation to the verb v, direct object of v, a prepositional relation to v, respectively. Their multi-argument events modeling allows to predict event is most likely happened at some point in the sequence. By counting the number of times occur event a_1 and a_2 in order to calculate the conditional probability $P(a_2|a_1)$ of seeing a_1 and a_2 in order. In order to infer events, they proposed a ranking candidate events a by maximizing $S(a) = \sum_{i=1}^{p-1} logP(a|a_i) + \sum_{i=p}^{|A|} logP(a_i|a)$, where $|A|$ is an ordered list of events, and p is the length of A. The model demonstrates that multi-argument events improves predictive accuracy of inferring held-out events. Like the same year, Modi and Titov (2014) presented another approach for event ordering tasks based on distributed representation (e.g., vectors of real numbers) of the event of predicates and their arguments and then the event representation was used in a ranker to predict the ordering of events. However, in this work, they concentrated on ordering tasks rather than predict missing events given a set of events. At the evaluation phase, their approach showed improvement in the $F1$ score on event ordering with respect to the graph induction approach Regneri et al. (2010) (84.1% vs. 70.6%). Unlike previous works rely on event counting-based methods, Modi (2016) proposed a neural network model based on compositional representations of events, namely *event embedding*, in order to predict missing event e_k in the prototypical ordering of events $(e_1, e_2, ...e_{k-1}, ..., e_n)$. The model demonstrated that it can obtain statistical dependencies between events in a scenario and outperformed count-based on the narrative cloze task.

More recent researches [Granroth-Wilding and Clark (2016); Pichotta and Mooney (2016); Hu et al. (2017)] work on multiple neural networks to improve the quality of the semantic properties of events were obtained by the model. These studies present an event embedding approach in terms of dense vector representation instead of symbolic event representation using by [Chambers and Jurafsky (2008); Regneri et al. (2010); Jans et al.

(2012)]. Pichotta and Mooney (2016) proposed a Long Short Term Recurrent Neural Networks (LSTM-RNN) to inferring held-out events from text and inferring novel events from the text. The input to the sequence modeling task is an event represents as a tuple of (v, e_s, e_o, e_p, p), where v is a verb lemma, e_s, e_o, e_p, p are arguments stands in subject, direct object, prepositional relations, prepositional relating v and e_p, respectively. The model coupled with Beam Search algorithm to generate event inferences on a given set of candidate events that exist in the training set. Granroth-Wilding and Clark (2016) introduced a compositional neural network to automatically acquire knowledge of event sequences from text, which provides a predictive model for use in narrative generation systems, by learning event embeddings using *Word2Vec* Mikolov et al. (2013) on narrative event chains, where each event is represented as a predicate word or an argument word. Hu et al. (2017) presented a contextual hierarchical LSTM can automatically generate a short text that describes a possible future sub-event from given previous sub-events that do not exist in the training data. However, all of these researches work on NLP from natural language text. Therefore, in order to work with scripts generation from sensors data, we either need to search for other techniques or find the relevant ways to connect sensors data to this domain.

2.2 State of the art in Human Activity Recognition from Wearable Systems

In this section, first of all, we present the definition of the problem of HAR in Section 2.2.1. In Section 2.2.2 we show HAR system including design issues and HAR methods, and the way this task is generally evaluated. In addition, conventional machine learning and deep learning on HAR systems are introduced in Section 2.2.3 and Section 2.2.4.

2.2.1 Human Activity Recognition Problems

In recent years, due to the significant development of micro-electronic and computer system allow people to interact with these devices as part of their daily living. An active research area was established with the main purpose is to extracting knowledge from collected data of these devices, namely *Ubiquitous Sensing* Perez et al. (2010). Particularly, human activity recognition is a task of this field and applied in several daily life domains such as medical, military and security. For instance, patients with diabetes, obesity, or heart disease are often required to follow a well-defined exercise routine as part of their treatment Jia (2009). Therefore, recognizing activities such as *walking, running*, or *resting* becomes quite useful to provide feedback to the caregiver about the patient's behavior.

As a consequence, one of the recent challenge approaches in our thesis applied machine learning to recognize human activities. In fact, human activity recognition (HAR) has been approached in two different ways, *namely external and wearable sensors*. External sensors are fixed in a predetermined point of interest, and the wearable sensors are attached to the user. My thesis is only concerned about wearable ones due to its advantages such as very comfortable for carrying, moving and allows to collect data for outdoors activities. Consequently, we could collect the data from the sensors (e.g., accelerometer, gyroscope, heart rate, breath rate, etc.) to detect human activities in different scenarios.

HAR is commonly solved using machine learning techniques. Similar to other machine learning applications, activity recognition requires two stages, i.e., *training* and *testing* (or *evaluation*). The training stage initially requires a time series dataset of measured attributes from individuals performing each activity. The time series are divided into time windows to apply feature extraction in order to filter relevant information in the raw signals and define metrics to compare them. Later, learning methods are used to generate an activity recognition model from the dataset of extracted features. Likewise, for testing, data are collected during a time window and a feature vector - also called feature set - is calculated. Such a feature set is evaluated in the a priori trained learning model, generating a predicted activity label. The data collected from wearable sensors are naturally indexed over the time dimension, which allows us to define the human activity recognition problem as follows.

Definition 1 (Human Activity Recognition Problem (HARP)) *Given a set of n time series* $S = \{S_0, \ldots, S_{n-1}\}$ *, each one from a particular measured attribute, and all defined within time interval* $I = [t_i, t_j]$*, the goal is to find a temporal partition* $(I_0, ..., I_{r-1})$ *of I, based on the data in S, and a set of labels representing the activity performed during each interval* I_k *(e.g., sitting, walking, etc.). This implies that time intervals* I_k *are consecutive, non-empty, non-overlapping, and such that* $\bigcup_{k=0}^{r-1} I_k = I$.

Definition 2 (Relaxed HAR problem) *Given a set* $W = \{W_0, \ldots, W_{m-1}\}$ *of m equally sized time windows, totally or partially labeled, and such that each* W_i *contains a set of time series* $S_i = \{S_{i,0}, \ldots, S_{i,k-1}\}$ *from each of the k measured attributes, and a set* $A = \{a_0, \ldots, a_{n-1}\}$ *of activity labels, the goal is to find a mapping function f:* $S_i \rightarrow A$ *that can be evaluated for all possible values of* S_i*, such that* $f(S_i)$ *is as similar as possible to the actual activity performed during* W_i.

Definition 3 *Given a classification problem with a feature space* $\chi \in \mathbb{R}^n$ *and a set of classes* $A = \{a_0, \ldots, a_{n-1}\}$ *, an instance* $x \in \chi$ *to be classified, and a set of predictions*

$P = \{p_0, \ldots, p_{k-1}\}$ for x, from k classifiers, the goal of a multi-classifier system is to return the correct label a^* iff $\exists p_i \in \mathbb{P} | a^* = p_i$

Definition 1 only enables one to recognize activities that are not performed simultaneously. Therefore, the definition 2 is solved for the case of co-occurrence activities, which performed more than one in a single time window. However, during transition windows is made the relaxation produces errors, thus the number of transitions is expected to be small as much as possible, which leads to deduce the relaxation errors for most applications. Moreover, there is no one algorithm can achieve the best accuracy for all activities. Hence, the multi-classifier should be considered as a problem in HAR, which is defined in definition 3.

2.2.2 Human Activity Recognition System

Design issue and human activity recognition methods

There are eight main challenges related to human activity recognition, namely (1) *definition of the activity set, (2) selection of attributes and sensors, (3) obtrusiveness, (4) data collection protocol, (5) recognition performance, (6) energy consumption, (7) processing, and (8) flexibility* Lara and Labrador (2013). In this work, we consider the primary issues mentioned below.

Firstly, the design of any HAR system depends on the activities to be recognized. In fact, changing the activity set A immediately turns a given HARP into a completely different problem. From the literature, seven groups of activities can be distinguished. These groups and the individual activities that belong to each group are summarized in Table 2.2. Secondly, the success of a HAR system depends also on the sensors and the attributes chosen to describe the data in a specific domain. In the literature, there are four groups of common attributes that are computed using wearable sensors in a HAR context: environmental attributes, acceleration, location and physiological signals Lara and Labrador (2013). The first place is the attributes that provide context information describes the individual's surroundings (e.g., temperature, audio level, etc.), the second place is triaxial accelerometers often used to recognize ambulation activity, the third place is Global Positioning System (GPS) currently equipped in cellular phones, the fourth place is the vital signs data (e.g., heart rate, respiration rate, skin temperature, etc.).

These attributes and sensors would give us a definition of the set of activities for the daily living domain. In fact, acceleration is one of the important attributes in our domain which would define set of activities (e.g., *walking, running, lying down, sitting, standing still, etc.),* while the environmental attributes could be interested (e.g., audio level and

Group	Activities
Ambulation	Walking, running, sitting, standing still, lying, climbing stairs, descending stairs, riding escalator, and riding elevator
Transportation	Riding a bus, cycling, and driving
Phone usage	Text messaging, making a call
Daily activities	Eating, drinking, working at the PC, watching TV, reading, brushing teeth, stretching, scrubbing, and vacuuming
Exercise/fitness	Rowing, lifting weights, spinning, Nordic walking, and doing push ups
Military	Crawling, kneeling, situation assessment, and opening a door
Upper body	Chewing, speaking, swallowing, sighing, and moving the head

Table 2.2: Types of activities recognized by state-of-the-art HAR systems

light intensity are fairly low allow to recognize that user might be *sleeping*), location or GPS enables to realize the places where users have been being can also be helpful to figure out their activities, and physiological signals are confidently a valuable attribute to improve the recognition accuracy, for instance, heart rate is at a high level could allow to observe that user might perform *running* rather than *sitting* or *lying*. Meanwhile, obtrusiveness and data collection protocol show that they have an effect on the accuracy of HAR systems to raise interesting questions such as how many sensors are enough and where are the places to attach sensors on a human body to improve the accuracy of human activity recognition task.

Regarding the design of a HAR system there are several aspects to deal with: (1) the features extraction methods, (2) the learning algorithm and (3) the quality of the training data.

Two approaches were introduced to extract the features from time-series data: *statistical and structural* are showed in Table 2.3. It depends on the given signal to choose relevant methods were illustrated in Table 2.4. For instance, [Chen et al. (2008b); He and Garcia (2009)] the acceleration signals are highly fluctuating and varying, therefore the statistical feature extraction - either time or frequency domain - is the best method to handle with these signals. Besides that, Lara et al. (2012) proposed that structural feature extraction - polynomial - is the best fit for the physiological signals such as heart rate, respiration rate, breath amplitude, and skin temperature.

Group	Methods
Time domain	Mean, standard deviation, variance, interquartile range (IQR), mean absolute deviation (MAD), correlation between axes, entropy, and kurtosis [Pärkkä et al. (2006), Tapia et al. (2007)]
Frequency domain	Fourier Transform (FT) [Bao and Intille (2004); Chen et al. (2008a)], Discrete Cosine Transform (DCT) Altun and Barshan (2010)
Others	Principal Component Analysis (PCA) He and Jin (2009), Linear Discriminant Analysis (LDA) Chen et al. (2008a), Autoregressive Model (AR), HAAR filters Hanai et al. (2009).
Linear	F(t) = mt+ b
Polynomial	F(t) = $a_0 + a_1 * t + ... + a_{n-1} * t^{n-1}$
Exponential	F(t) = $a * b^t + c$
Sinusoidal	F(t) = $a * sin(t + b) + c$

Table 2.3: Group of features extraction methods

Attributes	Features
Altitude	Time domain
Audio	Speech recognizer
Barometric pressure	Time domain and frequency domain
Humidity	Time domain
Light	Time domain and frequency domain
Temperature	Time domain

Table 2.4: Categories of attributes and features extraction methods

In general, these features extraction methods can influence on recognition accuracy along with the quality of data and learning algorithm. Moreover, it is believed that features extraction methods would take a significant effect on HAR classification accuracy, for instance, Khan et al. (2010) proposed a group of features can improve the recognition accuracy up to 97% by using Autoregressive model coefficient, Tilt Angle, Signal Magnitude Area. This work shows that carefully choosing features has a great impact on the classification, hence the need to explore new methods to extract features.

For learning algorithm, there are two mainstream learning approaches, namely supervised and unsupervised learning, which deal with labeled and unlabeled data, respectively.

Supervised learning is one of the most popular approaches to activity recognition with many algorithms as presented in Table 2.5.

Type	Classifiers
Bayesian	Naive Bayes and Bayesian Networks Lara et al. (2012)
Clustering	K-nearest neighbor Lara et al. (2012)
Neural Networks	Multilayer Perceptron [Kwapisz et al. (2011); Bayat et al. (2014)]
Kernel Method	Support Vector Machine [Bayat et al. (2014); Anguita et al. (2012); Khan et al. (2014)]
Fuzzy Logic	Fuzzy Basis Function and Fuzzy Inference System Berchtold et al. (2010a)
Regression methods	Multiclass Logistic Regression Riboni and Bettini (2011), Additive Logistic Regression Lara et al. (2012)
Markov Models	Hidden Markov Models and Conditional Random Fields Blachon et al. (2014)
Classifier ensembles	Boosting and Bagging Lara et al. (2012)

Table 2.5: Classification algorithms used by state of the art HAR systems

Evaluation metrics

When evaluating a machine learning algorithm, the training and testing datasets should be disjoint. This is with the aim of assessing how effective the algorithm is to model unseen data. A very intuitive approach is called random split and it simply divides the entire dataset into two partitions: one for training and the other one for testing - usually, two-thirds of the data are for training and the remaining one third is for testing. However, a random split is highly biased by the dataset partition. If instances in any of the training

or testing sets are concentrated in a particular feature space sub-region, the evaluation metrics would not reflect the actual performance of the classifier. Therefore, a more robust approach is cross-validation. In k-fold cross-validation, the dataset is divided into k equally-sized folds. In the first iteration, the very first fold is used as the testing set while the remaining k-1 folds constitute the training set. The process is repeated k times, using each fold as a testing set and the remaining ones for training.

In the end, the evaluation metrics (e.g., accuracy, precision, recall, etc.) are averaged out over all iterations. In general, the selection of a classification algorithm for HAR has been merely supported by empirical evidence. The vast majority of the studies use cross-validation with statistical tests to compare classifiers' performance for a particular dataset. The classification results for a particular method can be organized in a confusion matrix M_{n*n} for a classification problem with n classes. This is a matrix such that the element M_{ij} is the number of instances from class i that were wrongly classified as class $j \neq i$. The following values can be obtained from the confusion matrix in a binary classification problem:

- True Positives (TP): The number of positive instances that were classified as positive.
- True Negatives (TN): The number of negative instances that were classified as negative.
- False Positives (FP): The number of negative instances that were classified as positive.
- False Negatives (FN): The number of positive instances that were classified as negative.

The accuracy is the most standard metric to summarize the overall classification performance for all classes and it is defined as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

The precision - often referred to as positive predictive value - is the ratio of correctly classified positive instances to the total number of instances classified as positive:

$$Precision = \frac{TP}{TP + FP}$$

The recall, also called true positive rate, is the ratio of correctly classified positive instances to the total number of positive instances:

$$Recall = \frac{TP}{TP + FN}$$

The F-measure combines precision and recall in a single value:

$$F - measure = 2 * \frac{Precision * Recall}{Precision + Recall}$$

2.2.3 Conventional Machine Learning on HAR Systems

There are a lot of studies and reports about human activity recognition using wearable sensors in recent years. HAR systems can be divided up following two dimensions. By learning approach, namely *supervised* and *semi-supervised*; and by the response time constraint, namely *on-line* and *off-line*. The on-line approach provides immediate feedback on the performed activities [Lara and Labrador (2012); Anguita et al. (2013)]. The off-line approach either needs more time to recognize activities due to high computational demands or are intended for applications that do not require real-time feedback [Kwapisz et al. (2011); Ronao and Cho (2015); Lara et al. (2012)]. Figure 2.1 presents a flow diagram of using conventional machine learning approaches based on sensors for activity recognition. First of all, the raw data was obtained by some types of sensors (e.g., smart-phones, smart-watches, WiFi, Bluetooth, audio, etc.). Secondly, the features were extracted from the raw signal by statistic information such as mean and standard deviation, etc. Finally, these features were fed into machine learning models (e.g., kNN, SVM, HMM, etc.) so as to perform the recognition tasks.

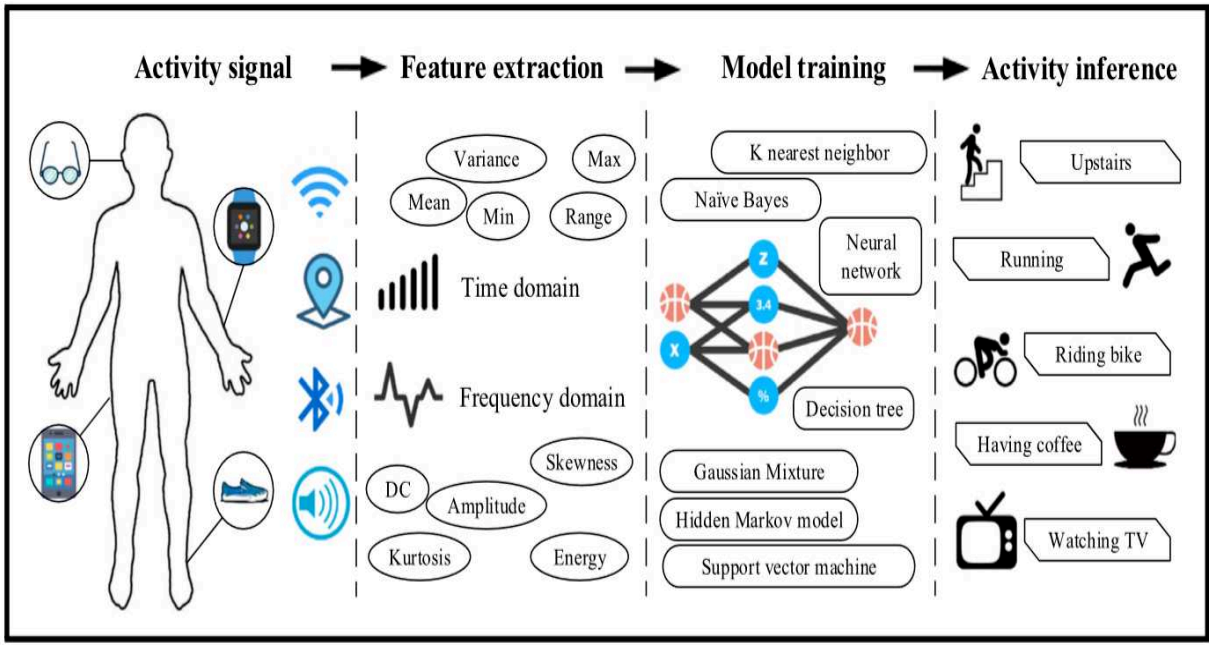


Figure 2.1: An Illustration of sensor-based activity recognition using conventional machine learning approaches Wang et al. (2019).

Supervised Online HAR system

Applications of online activity recognition systems are often seen in health-care, where the continuous monitoring of patients who have physical or mental pathologies is crucial for their protection, safety, and recovery. Therefore, it is required for an online system to

identify what the user is currently doing Lara and Labrador (2012). Likewise, interactive games or simulators may enhance the user's experience by considering activities and their locations Riboni and Bettini (2011).

Study	Sensors [1]	Activities [2]	Algos [3]	Performance
Berchtold et al. (2010b)	ACC (phone)	AMB, PHO (10)	RFIS	71-98% accuracy
Riboni and Bettini (2011)	ACC (watch, phone), GPS	AMB, DA (10)	COSAR	93% accuracy
Lara and Labrador (2012)	ACC, VS (Chest)	AMB (3)	C4.5	92.6% accuracy
Blachon et al. (2014)	ACC, Audio	DA (5)	RF, CRF	74% F-measure

Table 2.6: Summary of state-of-the-art in online HAR systems

[1] ACC: Accelerometers, VS: Various Sensors, HAR: Heart rate.

[2] AMB: Ambulation activities, DA: Daily activities, PHO: Activities related to phone usage.

[3] C4.5: Decision Tree, RF: Random Forest, CRF: Conditional Random Fields, RFIS: Recurrent Fuzzy Inference System.

Berchtold et al. (2010b) investigated a system, namely *ActiServ*, for smart-phone based activity recognition. The system enables to integrate labeled annotation with new training data directly on smart-phone at run-time, and using fuzzy inference to classify for ten ambulation activities based on phone's accelerometer feature extraction. The results showed that when using an online-algorithm the accuracy of recognition achieved only 71%, while offline-algorithm increased up to 98% in the case of subject dependent analysis.

Riboni and Bettini (2011) presented a framework namely COSAR for context-aware activity recognition using statistical and ontological reasoning under a mobile Android platform. In this work, the author introduced ontological reasoning along with a statistical method to recognize ten daily activities such as *brushingTeeth, hikingUp, hikingDown, ridingBycycile, jogging, standingStill, strolling, WalkingDownstairs, WalkingUpstairs, WritingonBlackBoard* that statistical methods can not classify alone. The idea is statistical inferencing is performed based on raw data retrieved from body-worn sensors (e.g., ac-

celerometers) to predict the most probable activities. Then, symbolic reasoning is applied to refine the results of statistical inferencing by selecting the set of possible activities performed by a user based on her/his current context. COSAR gathers data from two accelerometers, one in the phone and another on the individual's wrist, as well as from the cellphone's GPS. COSAR uses an interesting concept of potential *activity matrix* to filter activities based upon the user's location. For instance, if the individual is in the office, he or she is probably not cycling. Another contribution is the statistical classification of activities with a historical variant. For example, if the predictions for the last five time windows were jogging, jogging, walking, jogging, jogging, the third window was likely a misclassification (e.g., due to the user performing some atypical movement) and the algorithm should automatically correct it. However, this introduces an additional delay to the system's response, according to the number of windows analyzed for the historical variant. The overall accuracy was roughly 93% using Multiclass Logistic Regression though, in some cases, *standing still* was confused with *writing on a blackboard*, as well as *hiking up* with *hiking down*.

Lara and Labrador (2012) proposed Vigilante, a mobile application for real-time human activity recognition under the Android platform. This application used accelerometers and physiological signals (e.g., heart rate, respiration rate, etc.) in order to recognize five ambulation activities. The feature extraction methods statistical and time, frequency domain was used to extract 84 features. Then, several learning algorithms were applied to classify activities such as Bayes Network, C45, MLP, SMO, etc. The author reported that C45 decision tree had the best performance by achieving up to 96.8% accuracy. However, the author also noticed that the models were extremely influenced by the dataset. When they tried to train with a given user and tested it on another user data, the classification accuracy significantly decreased to 64%.

Blachon et al. (2014) introduced another online HAR system using the mobile Android platform. They reported that the audio signal can play an important role with accelerometers to recognize nine ambulation activities. The time and frequency domain were used to extract 74 features. Later, learning algorithms such as C45 decision tree, Random forest, and Conditional Random Fields were applied to classify the activities. The author reported that the Random Forest was the most performance with an overall F-measure of 74%.

As we can see from the above online HAR systems, it depends on the application requirement so that the appropriated approach can be selected. For instance, COSAR can provide a broader set of activities to be recognized, therefore COSAR should be considered in case of large activities recognition. Moreover, Vigilante is only approach to integrate vital signals with accelerometers to allow recognize activities related to the health-care purpose.

Supervised Offline HAR system

Study	Sensors [1]	Activities [2]	Algos [3]	Performance
Zhu and Sheng (2009)	ACC (wrist, waist)	AMB, TR (12)	HMM	90%
Kwapisz et al. (2011)	ACC	AMB(6)	MLP, C4.5, LR	MLP: 91.7%
Lara et al. (2012)	ACC, VS (Chest)	AMB (5)	ALR, Bagging, C.4.5, NB, BN	95.7%
Bayat et al. (2014)	ACC	DA (6)	MLP, LB, SVM, RF	91.15%
Saputri et al. (2014)	ACC	AMB (6)	ANN	93%
Sztyler and Stuckenschmidt (2016)	ACC	AMB (8)	C4.5	89%

Table 2.7: Summary of state-of-the-art in offline HAR systems

[1] ACC: Accelerometers, VS: Various Sensors.

[2] AMB: Ambulation activities, DA: Daily activities, TR: Transition between activities.

[3]C4.5: Decision Tree, LR: Logistic Regression, ALR: Additive Logistic Regression, MLP: Multilayer Perceptron, NB: Naive Bayes, SVM: Support Vector Machines, RF: Random Forest, LB: LogitBoost, HMM: Hidden Markov Model.

Unlike online systems, offline HAR systems are not dramatically affected by processing and storage issues because the required computations are performed in a server with large computational and storage capabilities. Additionally, energy expenditures are not analyzed in detail as a number of systems require neither integration devices nor wireless communication so the application lifetime would only depend on the sensor specifications.

Ambulation activities are recognized very accurately by [Lara et al. (2012); Khan et al. (2010)]. These systems place an accelerometer on the subject's chest, which is helpful to avoid ambiguities due to abrupt corporal movements that arise when the sensor is on the wrist or hip. An additional challenge was raised in Bao and Intille (2004), where activities such as eating, reading, walking, and climbing stairs could happen concurrently. However, no analysis is presented to address that matter. In the following paragraph, we introduce several HAR systems that use an off-line supervised learning approach.

Kwapisz et al. (2011) presented a HAR system under the mobile Android platform. This system only used the accelerometer to recognize six ambulatory activities such as

walking, jogging, climbing up stairs, climbing down stairs, sitting and standing from twenty-nine users attached in their pockets. 43 features were extracted from the time-domain method. Decision tree C4.5, Multilayer Perceptron (MLP) and Logistic Regression were applied to classify the activities. The author reported that MLP can obtain 91% accuracy outperforming the other learning algorithms. However, the activities climbing up and climbing downstairs were the most difficult to recognize.

The system proposed by Zhu and Sheng (2009) uses Hidden Markov Models (HMM) to recognize ambulatory activities. Two accelerometers, placed on the subject's wrist and waist, were connected to a PDA via a serial port. The PDA sent the raw data via Bluetooth to a computer that processed the data. This configuration is obtrusive and uncomfortable because the user has to wear wired links that may interfere with the normal course of activities. The extracted features are the angular velocity and the 3D deviation of the acceleration signals. The classification of activities operates in two stages. In the first place, an Artificial Neural Network discriminates among stationary (e.g., sitting and standing) and non-stationary activities (e.g., walking and running). Then, an HMM receives the ANN's output and generates a specific activity prediction. An important issue related to this system is that all the data were collected from one single individual, which does not permit to draw strong conclusions on the system flexibility.

Lara et al. (2012) proposed a system that combines acceleration data with vital signs to achieve accurate activity recognition. Centinela recognizes five ambulation activities and includes a portable and unobtrusive real-time data collection platform, which only requires a single sensing device and a mobile phone. Time- and frequency-domain features are extracted from acceleration signals while polynomial regression and transient features are applied to physiological signals. After evaluating eight different classifiers and three different time window sizes, and six feature subsets, Centinela achieves an overall accuracy of over 95%. The results also indicate that incorporating physiological signals allows for a significant improvement of the classification accuracy. As a trade-off, Centinela relies on ensembles of classifiers accounting for higher computational cost, and it requires wireless communication with an external sensor, which increases energy expenditures.

Bayat et al. (2014) introduced a recognition system based on acceleration data, which used a new low-pass filter to detach the component of gravity acceleration from body acceleration in raw data. There are 18 features were selected from 24 features extracted along with a window size of 128 samples and 50% overlap. Six daily activities were classified in several classification models such as MLP, LB, SVM, RF. The achieved accuracy by an average of these models was 91.15%.

Saputri et al. (2014) proposed a neural network with a three-stage genetic algorithm-based feature selection to solve for the flexibility or user-independent issue in HAR. Time-

domain features (e.g., mean, root mean square, variance, correlation, and standard deviation, etc.) were extracted by a window-size 100 samples. Then, these features were selected by a genetic algorithm, so that it can gain attention on the feature set, which is appropriated for different activities for a single person and effective in representing these activities across multiple subjects. The system shows that it can achieve 93% of average accuracy for subject-independent in HAR.

Sztyler and Stuckenschmidt (2016) presented an approach for activity recognition based on acceleration data, which located in different on-body positions with multiple wearable devices. This work concentrated on both subject-independent and subject-dependent in HAR. Without the information of the device position, the classifiers (e.g., Decision Tree) only can achieve 80% of F-measure. While with information on device position the classifier can obtain higher F-measure up to 89%.

2.2.4 Deep Learning on HAR Systems

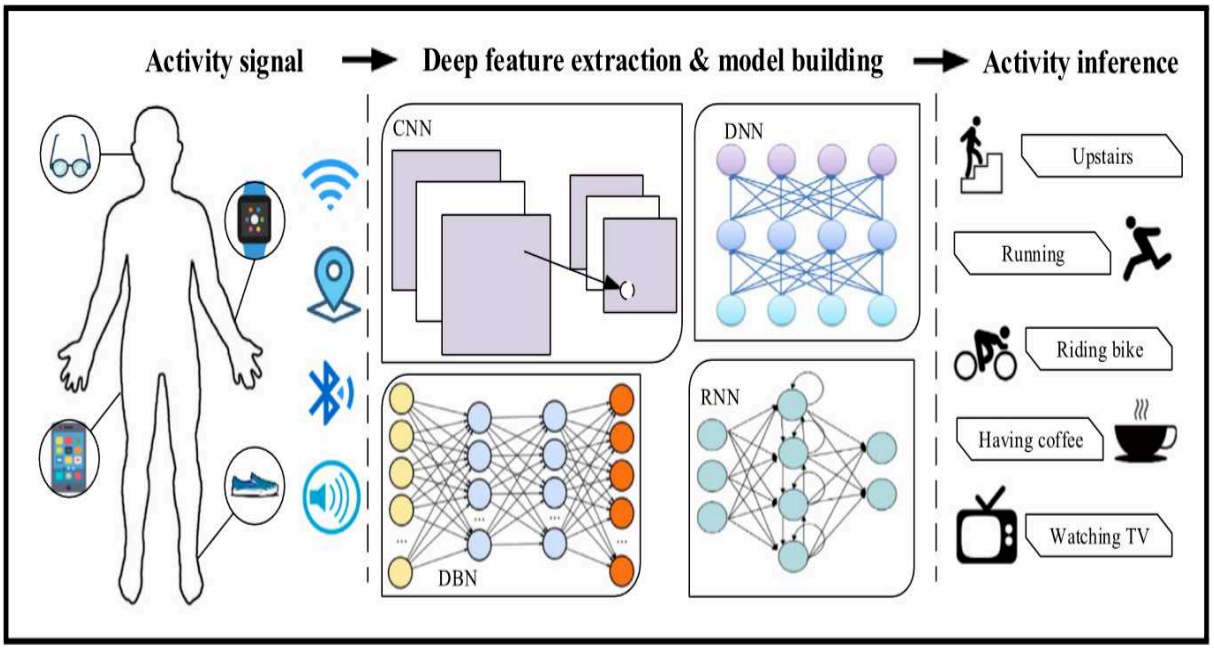


Figure 2.2: An Illustration of sensor-based activity recognition using deep learning approaches Wang et al. (2019).

Many traditional HAR learning algorithms like Decision Tree, Support Vector Machine [Bayat et al. (2014); Anguita et al. (2012); Khan et al. (2014)], Conditional Random Forest Blachon et al. (2014), Naive Bayes, and hidden Markov models Zhu and Sheng (2009), K-nearest neighbor Sani et al. (2017), etc. have been proposed to achieve good accuracy on human activity recognition task in the few decades. However, there are several shortcomings to traditional HAR methods as mentioned in Wang et al. (2019) such as (1)

the features are often extracted by heuristic and hand-crafted feature extraction methods; (2) shallow features (e.g., mean, variance, frequency, etc.) usually are only able to be used for low-level activity recognition task (e.g., walking, jumping, etc.). Nevertheless, it is hard to use for recognizing high-level or context-aware activities Yang (2009) (e.g., drinking coffee, have breakfast, etc.); (3) traditional learning approaches often need a large number of labeled data for training model. To tackling these limitations, in recent years there are many types of researches interest in deep learning for HAR issues [Jiang and Yin (2015a); Ronao and Cho (2015); Murad and Pyun (2017); Ignatov (2018); Bevilacqua et al. (2018)]. Figure 2.2 shows a demonstration of using deep learning approaches based on sensors for activity recognition. As we can see from the flowchart of Figure 2.2, a difference compares to Figure 2.1 (conventional machine learning) is the feature extraction and model building procedures are usually carried out at the same time, it is also an advantage of deep learning approaches in comparison with conventional machine learning approaches. Therefore, manually designed features are replaced by features learning automatically through the network. Moreover, deep generative models Hinton et al. (2006) allows exploiting the unlabeled samples for the training model. In addition, training on a large-scale labeled dataset using deep learning models can be transferred to new tasks where there are few or none labels. In the following paragraph, we investigate the deep learning models used in HAR tasks (cf. Table 2.8)

Model	**Description**
DNN	Deep fully-connected network, artificial neural network with deep layers
CNN	Convolutional neural network, multiple convolutional operations for feature extraction
RNN	Recurrent neural network, network with time correlations and LSTM
DBN/RBM	Deep belief network, restricted Boltzmann machine
SAE	Stacked auto-encoder, feature learning by decoding-encoding auto-encoder
Hybrid	Combination of some deep models

Table 2.8: Deep learning models for HAR systems surveyed by Wang et al. (2019)

Deep neural network

Artificial neural network (ANN) is a fundamental network for developing the deep neural network (DNN). It is designed to contain more layers than ANN (with very few hidden

layers), thus DNN is able to learn from a large amount of data. Vepakomma et al. (2015) presented a wristocracy framework that allows recognizing 22 fine-grained activity contexts with various activity classes (e.g., sitting on a sofa, use the refrigerator, walk indoor, etc.). The hand-engineered features were extracted by a sliding windows of 2 seconds from accelerometers, gyroscope and location context (Bluetooth beacon message), then these features are fed into a DNN model. Hammerla et al. (2016) explored a deep feed-forward network (DNN) and compare with other networks such as Convolutional neural network (CNN) and Recurrent neural network (RNN). These networks performed HAR task on three representative datasets that contain movement data captured with wearable sensors. As a result, these models outperformed the state of the art on benchmark datasets.

Convolutional neural network

In recent years, CNN is applied in many areas such as image classification, speech recognition, and text analysis because of its automatic feature extraction from signals and promising results on the tasks. In time series classification like HAR, CNN has two different models: local dependency (correlated) and scale in-variance (frequencies) as mentioned in Wang et al. (2019). Most researches on CNN for HAR models concentrated on several aspects as follows: *input adaptation*, *pooling*, and *weight-sharing*.

- Input adaptation: The input adaptation is the approach to form sensor readings (e.g., 3-axis accelerometers) in HAR to be the same as a virtual image, which is the inputs of CNN. There are two types of adaptation in HAR: (1) *data-driven* treats each dimension sensor readings as a channel, then applies 1D convolution on them [Zeng et al. (2014a); Jiang and Yin (2015a)]. Thereby, the drawback of this approach is the ignorance of dependencies between dimension and sensors, which may impact on the performance as mentioned in Wang et al. (2019); (2) *model-driven* reshapes the inputs to a virtual 2D image in order to adopt 2D convolution [Ha and Choi (2016a); Jiang and Yin (2015b)]. Hence, this approach can improve the temporal relation of the sensor. However, the formulation of the image from the time series is a non-trivial task.

- Pooling: An interest of pooling is it can accelerate of training process on large data Bengio (2013). Most approaches performed average and max pooling after convolution [Ha and Choi (2016a); Kim and Toomajian (2016)].

- Weight-sharing: this approach can also speed up the training process on a new task [Zebin et al. (2016); Zeng et al. (2014a); Ha and Choi (2016b)].

Auto-encoder

Auto-encoder (AE) is a feed-forward neural network, which is similar to MultiLayer Perceptron (MLP). Figure 2.3 is shown an AE architecture, which comprises of two phases: (1) encoder phase is the layers contain input layer and hidden layer; (2) decoder phase is the layers consist of hidden layer and output layer. The encoder phase can be formulated as Eq. 2.1, where W^1 is the weight matrix and b^1 is the bias for the encoder phase. The decoder phase can be defined as Eq. 2.2, where W^2 is weight matrix and b^2 is the bias for the decoder phase. It aims to learn more advanced latent feature representation through an unsupervised learning schema Chen et al. (2018).

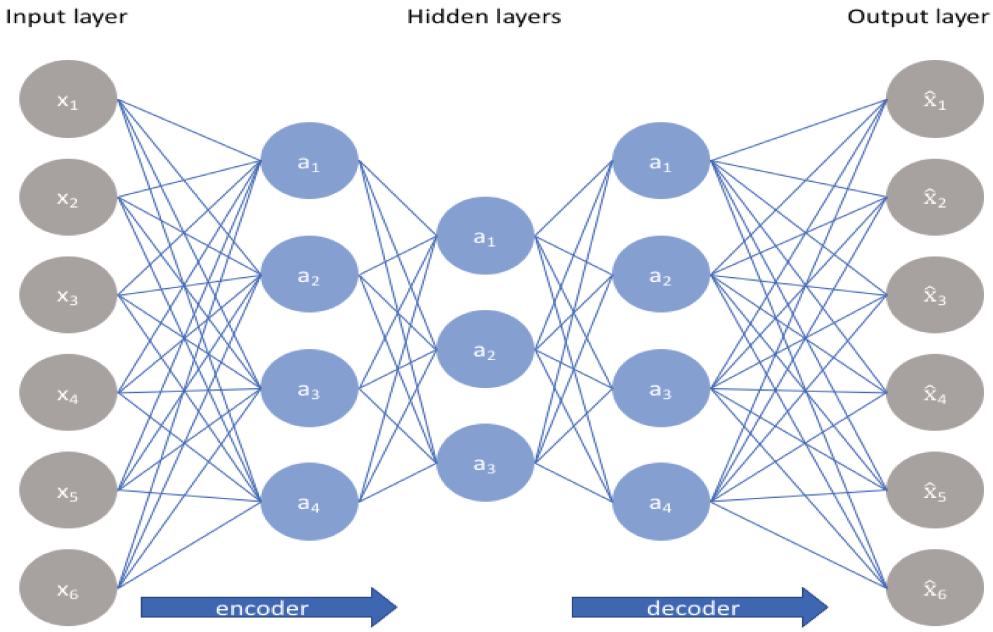


Figure 2.3: Architecture of an auto-encoder.

$$h(x) = f(W^1 x + b^1) \tag{2.1}$$

$$\widehat{x} = f(W^2 x + b^2) \tag{2.2}$$

Figure 2.4 is illustrated as a stacked auto-encoder architecture. SAE is a hierarchical model comprises of multiple auto-encoders Vincent et al. (2008). Therefore, SAE is suffered from poor local optimum problems during the whole network training process. To solve this problem, SAE trains on each layer as a fundamental model of auto-encoder. After several stages of training, the learned features are stacked with the output layer or classifier layer on top SAE to create a classifier. The effective performance of unsupervised feature learning is an advantage of SAE. Therefore, there are several researches work on SAE for HAR [Chen et al. (2018); Wang (2016); Li et al. (2014)]. Li et al. (2014)

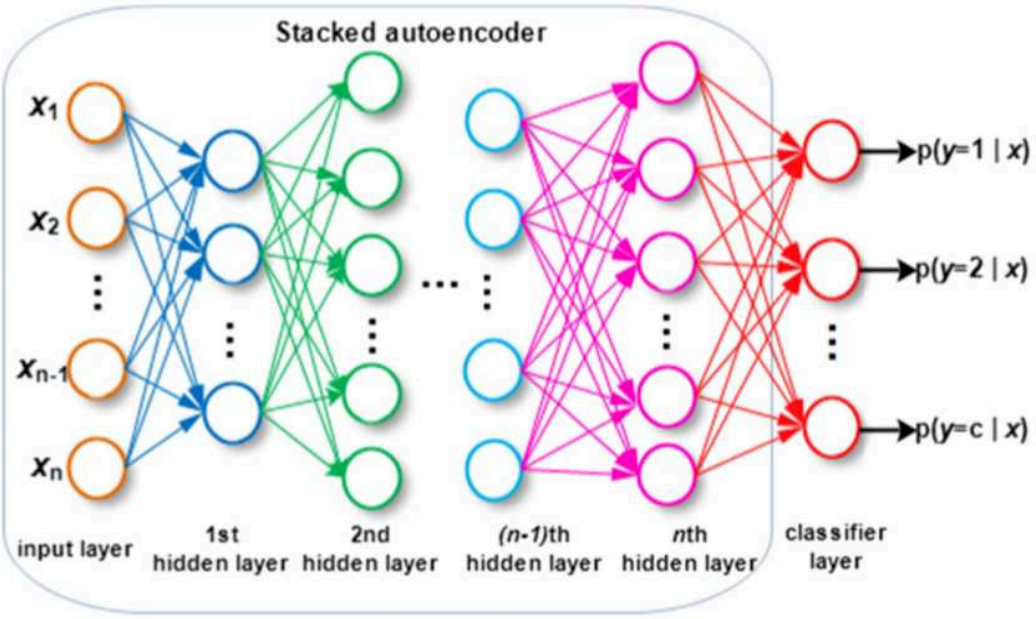


Figure 2.4: Architecture of a stacked auto-encoder Chen et al. (2018)

proposed a sparse auto-encoder by using Kullback-Leibler divergence and noise into the cost function, which could enhance the performance of HAR. In addition, Wang (2016) investigated in continuous auto-encoder (CAE) in order to extract the feature of non-linear data by adding Gaussian random units into activation function, and a novel fast stochastic gradient descent (FSGD) so as to reduce the training time for the model. Chen et al. (2018) introduced the greedy layer-wise scheme in pre-training, then performs the fine-tuning. However, due to the dependence on the complexity of layers and activation functions, SAE might be hard to look for optimization as mentioned in Wang et al. (2019).

Restricted Boltzmann machine

Restricted Boltzmann machine (RBMs) is a generative stochastic artificial neural network, which is restricted in their neuron and forms a bipartite graph. It is composed of two layers; i.e., visible and hidden Hinton et al. (2006) where each neuron in the layer is not connected. The hidden layer assists to forward pass or classification task and the visible layer helps for backward pass/reconstruction of the original input (known as generative learning). A deep belief networks (DBNs) is defined as a stack of RBMs, in which two consecutive layers are represented as an RBM. This stack is often followed with a softmax layer to create a classifier or help to cluster unlabeled data in an unsupervised task.

In recent years, there are several researches work on HAR using RBMs and DBNs. Zhang et al. (2015) introduced a real-time application for activity recognition on smart-phones using DBNs. In pre-train, the input to DBNs is acceleration data and forms a Gaussian-binary RBMs between input layer and first hidden layer. After pre-train, a soft-

max layer is added to classify seven activities (e.g., walking, running, lying , etc.). The DBNs classification model achieved the highest accuracy in comparison with other learning models such as Bayesian nets, SVM, LR, J48, RF, and KNN. Hammerla et al. (2015) proposed a system evaluates Parkinson's Disease (PD) in naturalistic environments (e.g., the daily life of individual affected by PD). A total of approximately 5500 hours of accelerometer data was collected by 34 participants in a home environment, in order to detect for 4 classes: *asleep, on , off* and *dyskinesia*. From the raw signal 91 hand-crafted features were extracted in each minute. Then, in the first step of training, the features are normalized and fed into RMBs to reconstruct the input features. In the second step of training, a next Gaussian-binary RBMs layer was taken the activation probabilities of the first RBMs as its input data. Finally, a softmax layer was added as a top-layer to classify the four classes.

Recurrent neural network

Recurrent neural network (RNN) is a class of artificial neural networks, where each neuron connected to create a directed graph along a temporal sequence. Long-Short-Term-Memory (LSTM) is an artificial RNN architecture, which deals with explode and vanishing gradient problems encountered in RNN. In recent years, there are few researches on RNN and LSTM for HAR tasks [Hammerla et al. (2016); Edel and Koppe (2016); Murad and Pyun (2017)]. For instance, Edel and Koppe (2016) presented a binarized BLSTM-RNN system, which re-defined all parameters (e.g., weight, input, output, hidden layers, etc.) in the network in term of binary values. Hence, the model only computed on bite-wise operation instead of using arithmetic operations. As a result, the system reduced the high computational cost of traditional RNN-LSTM, while it still remains a good accuracy for the HAR task. Murad and Pyun (2017) proposed a deep recurrent neural network (DRNN), which is able to capture long-range dependencies in variables-length input sequences. Their experiments showed that DRNN outperforms other classifications such as DBNs, CNN, kNN, SVM in benchmark datasets.

Hybrid model

A hybrid model is the connection of different types of deep learning models together such as CNN and RNN [Morales and Roggen (2016); Yao et al. (2017)]. For instance, Yao et al. (2017) proposed a deep learning framework, namely DeepSense, which combines convolutional neural network and recurrent neural network so as to perform three tasks: car tracking with motion sensors, human activity recognition and user identification with biometric motion analysis. It is shown that CNN is responsible for capturing the spa-

tial relationship, and RNN can achieve to obtain for the temporal relationship. Hence, integration of CNN and RNN can boost the activity recognition task that have varied time span and signal distribution. Similarly, Morales and Roggen (2016) also proposed a framework (DeepConvLSTM) based on convolutional and LSTM neural networks, which allowed to perform sensor fusion naturally and automatically extracted features using CNN, after that LSTM was used for learning the temporal dynamic of feature activation. As a result, DeepConvLSTM was compared with deep non-recurrent networks and previous learning models on two well-known public HAR datasets (OPPORTUNITY, Skoda). The result demonstrated that DeepConvLSTM outperformed these learning models. Other researches integrate CNN with other models such as SAE Zheng et al. (2016) and RBM Liu et al. (2016). In these works, CNN performed features extraction and generative models assist to adjust the input data for the training process. There is an expectation to expanded researches for this area in the future.

In summary, although this state-of-the-art of the HAR domain shows that activities daily living can be recognized with conventional and deep machine learning. However, there is still no approach that allows generating a story or script from the HAR system. In addition, conversely, there is no approach for script generation on the NLP domain using sensor-based human activity recognition. In this thesis, we approach these issues in order to connect both domains together. Moreover, we will deal with existing issues on each domain such as imbalanced data and high-level activities recognition, which enhance the prediction of the HAR system. Besides, we handle issues on script generation such as event representation, the sequence of events classification so as to predict the next event, missing event in natural language text on the NLP domain.

Chapter 3

Problematic and Overall approach

3.1 Problem Statement

As stated in the introduction, generating a coherent textual story from raw sensor data is highly challenging. From a pure *data abstraction* and *data processing* point of view, the following research questions should be addressed.

1. What information must be extracted from ambient data to be able to generate a story?
2. What data processing approach would make the extraction of this information possible?
3. How this information should be structured?
4. What Natural Language Generation process must be employed to generate a story from this set of information?

However, from a *cognitive* point of view, the problem is not defined in terms of data processing but in terms of information communication. Hence, in this specific case, the problem of story generation can be seen as constructing a discourse that respects the following objectives.

1. Linguistically represents a set of events discernible in the data.
2. Relates them temporally and causally.
3. Integrates these chunks of the information under a common communicative purpose. This communicative purpose will drive, among others, the below sub-objectives
   (a) The motivation and/or justification of decisions/actions in the story;
   (b) The filtering/amplification of pieces of information.

3.2 A Data Abstraction Approach

From a data abstraction point of view, the story is about what people do and what they feel during daily life. Hence, in order to generate a story, we need to know what kinds of activity are of interest, and what are the methods to automatically extract them.

In recent years with the advanced technology of smart-phone sensors (e.g., capacity increase, cost efficiency, power efficiency, etc.) it became possible to use it for human activity recognition (HAR) purposes [Bayat et al. (2014); Kwapisz et al. (2011); Anguita et al. (2013)]. Indeed, HAR plays an important role in automatically story generation that allows us to automatically identify and discover *what* and *when* people are doing during their daily life. Moreover, Recognizing Human Activity from sensors data can assist human to predict physical activity in everyday life such as *walking*, *running* and *working at a computer*, etc. [Lara and Labrador (2013); Bayat et al. (2014); Blachon et al. (2014); Kwapisz et al. (2011)]. Therefore, HAR is a core requirement in this thesis in order to generate a story.

In HAR, the model is typically acquired using machine learning over datasets collected from smartphones and manually labeled. Such a model should be invariant to the users and devices on which the data is collected Lara and Labrador (2013). To permit the model to generalize, a large amount of data from several different users and devices is necessary. Another important problem is the **class imbalance**. Indeed, data-sets acquired from real-life do not contains uniformly distributed examples. For instance, the *jump* activity might far less present in the data than the *sitting* activity. Such a situation biases the learning towards the majority classes. Thus, a solution must be found to make unbiased learning possible. To address the HAR problem, we will use Artificial Neural Network models learned on smartphone data and propose a meta-learning framework to deal with imbalanced data.

However, most of HAR systems in recent researches [Anguita et al. (2012); Khan et al. (2014); Blachon et al. (2014); Nguyen et al. (2018)] aims at recognizing discrete human activity labels. As mentioned in Miyanishi et al. (2018) it is not sufficient to generate a story with discrete human activity labels. Indeed, HAR only answers the question of the *what* and *when* but not *where*. Furthermore, conventional learning relies on carefully handcrafted features. However, such features might be sub-optimal for the task. Hence, we will use recent deep learning techniques to acquire HAR models that are able to work directly on **raw signal data** and which can predict both the **activity and localization** of the user.

Regarding **text generation**, as introduced in Chapter 2, There have been researches for script/story generation from text in Natural Language Processing [Modi and Titov (2014); Modi (2016); Chambers and Jurafsky (2008); Regneri et al. (2010); Frermann

et al. (2014); Harrison et al. (2016)] or data-to-text generation in Natural Language Generation [Portet et al. (2009); Reiter and Dale (2000); Reiter et al. (2005); Miranda (2018)]. However, to the best of our knowledge, there is no approach of story generation from wearable sensor data using machine learning techniques, because sensors data has very limited information. To deal with the generation, we adopt a simple concept-to-text approach Qader et al. (2018) using a sequence-to-sequence deep model. In this approach, the information to be communicated is represented as a linearized Abstract Meaning Representation which is feed to a Recurrent Neural Network which in turn outputs a sequence of words representing the story. Such a model has proved to be very effective in recent end-to-end generation tasks Dušek et al. (2018).

3.3 The Cognitive Side of the Approach

Regarding the structuring aspect of a story, there is a large number of ways to perform it according to the audience, the communication goal and even the medium with which the story is being expressed (e.g., picture, video, text). Since the objective is to make the generated story highly understandable by a human, we take a cognitive approach to the story by considering the *Scripts* theory. Indeed, much of our common sense knowledge about the world (such as what usually happens when going to a restaurant) is thought to be represented in our mind by *Scripts*. *Script theory* was introduced by Tomkins (1978) who claimed that human behavior mostly follows patterns called *Scripts* which are stereotypical likely sequences of events. This theory has been popularized in AI by Schank and Abelson (1975) who introduced it as a method for representing procedural knowledge. Hence, in this thesis, we define an *event/narrative* as a sentence of words that describes human activities, action, and their location. We call *script* as a sequence of event/narrative, which is represented in the temporal order. Finally, we consider the *story* as a sequence of scripts.

However, **script models** must be acquired to be useful for story generation. If scripts are built by humans from experience, computers extract them from data. Over the past decade, there have been plenty of studies [Modi et al. (2016); Modi (2016); Modi and Titov (2014); Frermann et al. (2014); Regneri et al. (2010); Granroth-Wilding and Clark (2016); Hu et al. (2017)] carried out on learning *scripts* using Natural Language Processing (NLP) but there is a lack of research regarding script learning from sensor data. Hence, to extract the script model, we address the issue by learning the script of daily life from the text with the hope they can be transferable to story generation from data. Once again, we will use a deep learning approach to this problem.

3.4 Proposed framework

In this research, we aim to explore and solve the problems mentioned in the above sections (e.g., imbalanced data in HAR, *what*, *when* and *where*, text generation, script learning) to automatically generate a story from the human activity using sensor data. In order to do this, we have to develop a system with the overall approaches as depicted in the following diagram 3.1.

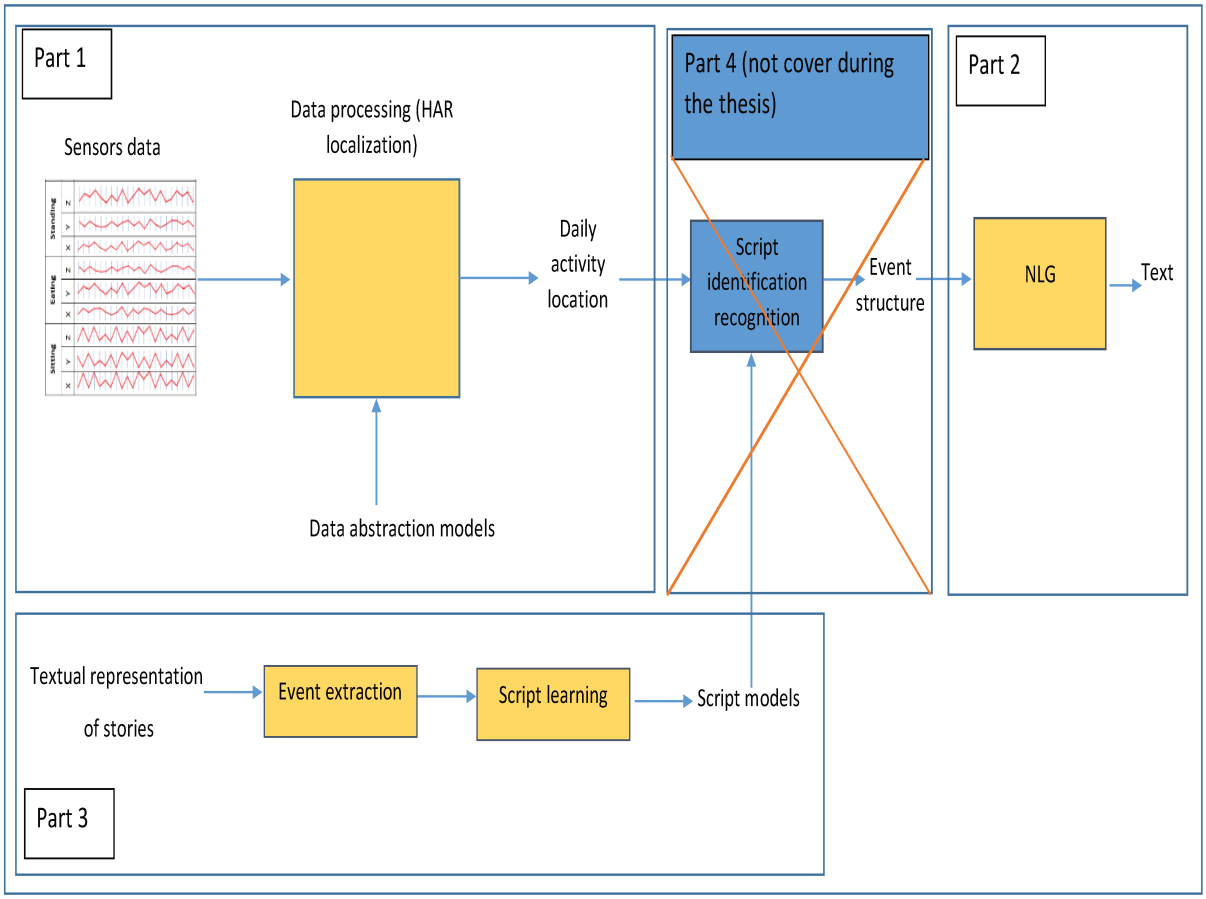


Figure 3.1: An overview of the story generation framework from sensor data with 4 main parts including Activity Recognition; Structured event generation; Text generation and Script learning. Structured event generation is not covered in this thesis

As we can see from Figure 3.1, there are four main parts describe the overall approach in order to achieve story/scripts generation from the human activity sensor data.

In the first part of the Figure 3.1, we used a window of time to extract the features from sensor data (e.g., accelerometers, audio, gyroscope, etc.) and feed the features as the input into a learning model (e.g., neural network) to perform the activity recognition task as previous researches on HAR [Lara and Labrador (2013); Bayat et al. (2014); Blachon et al. (2014); Kwapisz et al. (2011)]. In order to enhance the accuracy of the recognition task, we developed a framework to deal with **imbalanced data** Nguyen et al. (2018).

Moreover, as reported in Yang et al. (2015) most existing work on HAR relies on heuristic hand-crafted feature design and shallow feature learning architectures, which cannot find those distinguishing features to accurately classify different activities. Therefore, Yang et al. (2015) motivate to develop an automate systematic feature learning from the raw signals as inputs to be fed into a deep convolution neural network, so that the feature learning and classification are mutually enhanced by unifying in one model. Moreover, deep architecture can obtain specific *variance* or salient patterns of signals at different scales. In addition, with this model, we can solve the problem of **temporal variation of ADL events** by removing repetitive events based on creating new segmentation using the time-slices on the raw signal. As a result, we can re-segment the events located on the appearance frequency of their labels in the raw signal. As mention in Miyanishi et al. (2018) simple ADL labels prediction such as *walking, running, etc.* is the shortcoming to generate a story or event time-line. Therefore, in this research, we also follow the approach in Miyanishi et al. (2018) by recognizing the sequence of combinations of semantic concepts (e.g., human physical activity, action, location).

In the second part of the Figure 3.1, the output of prediction of HAR (the sequence of combinations of semantic concepts) will be considered as the input of sequence to sequence neural network model Sutskever et al. (2014b), so that the system can generate the candidate event/text descriptions from the semantic concepts. In fact, generating descriptions is a popular research in visual content (e.g., images and video) can be divided into four different approaches: (1) generating descriptions for images or videos contain some associated text [Aker and Gaizauskas (2010); Feng and Lapata (2010)], (2) generating descriptions by using manually defined rules or templates [Tan et al. (2011); Guadarrama et al. (2013)], (3) retrieving existing descriptions from similar visual content [Farhadi et al. (2010); Ordonez et al. (2011)], (4) learning a language model from a training corpus to generate descriptions [Kulkarni et al. (2011); Kuznetsova et al. (2012)]. In order to generate text descriptions from semantic concepts, we adopt these approaches by combining approach (2) and (4). At first, we generate the text description corpus from semantic concepts by simple detection rules as Tan et al. (2011) to glue the gesture, action, and location. However, using manually defined rules limit the natural flexibility of language as mentioned in Kuznetsova et al. (2012). Therefore, in the next step, the text description corpus can be considered as a target, the semantic concepts are determined as the source for the sequence to sequence neural network (seq2seq) model. By using the seq2seq model we can generate and enrich the semantic and syntactic of the generating candidate event/text descriptions. By recognizing the sequence of combinations of semantic concepts, we can obtain a sequence of candidate event/text descriptions (*Event timeline/Scripts*) that were generated from the seq2seq model.

In the third part of Figure 3.1, we aim to build a method to represent events and learning model from the natural text, which allows predicting the next event or missing event from a sequence of events, called script modeling. Indeed, in recent years many researches have been carried out to study the *Scripts*, which is the stereotypical sequence of events in prototypical scenarios, was first introduced by Schank and Abelson (1975). A script system not only predicts the next events from narrative text Granroth-Wilding and Clark (2016), but it also can assist to infer the missing event from events explicitly extracted from the natural text [Modi (2016); Modi et al. (2016)]. An example of a "bus taking" scenario is shown in Figure 6.1, where events are partially ordered or can be flexible. In fact, typical use of event chain knowledge is to help infer what is likely happen next given the previous event sequences in a scenario. For instance, as can be seen from the left side of Figure 3.2 the reader is expected to infer that the narrator could have been "X get off bus" given the text "X notice intended stop is next". Since the inputs are the different scenarios (e.g. scenario 1 and scenario 2) on the right side of Figure 3.2.

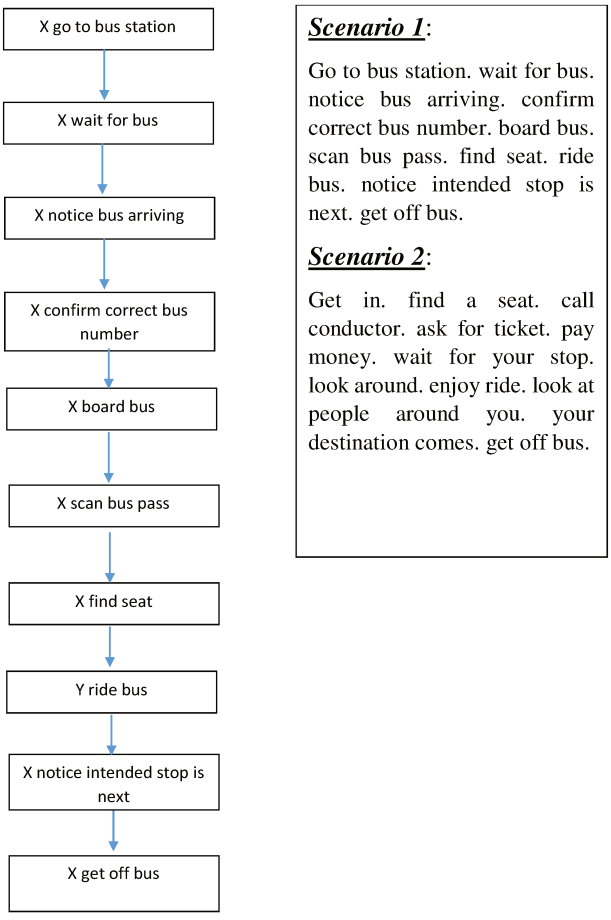


Figure 3.2: Events sequence of bus taking

Moreover, this kind of system can apply to many domain of artificial intelligence, more specifically on automatic narrative generations systems such as question answering, information extraction, and discourse understanding. In fact, an automatic narrative generation

system is a problem to automatically construct the structure of some level of abstraction such as sequences of events, actions or protagonist and entities that participate in that can be told as a story.

The very first devoted work by Schank and Abelson (1975) studied script knowledge focused on manual construction of its bases. Then in the past decade, there were many researches revised this work, which interested in automatically induction of script knowledge from text such as: [Chambers and Jurafsky (2008); Pichotta and Mooney (2014)] carried out studies on naturally occurring texts or Regneri et al. (2010) worked on crowd-sourced data. Most of these approaches focused on count-based techniques that represent events as a graph. Each node of the graph is an event, which is modeling as a verbal predicate with tuples of its arguments, and arcs correspond to the temporal precedence relation. Consequently, these methods usually suffer from the data sparsity because of the limitation of their model to capturing dependencies only between events have the common entities (e.g. protagonist) and the probability of predicate and argument estimates based on their co-occurrences (count-based method). Furthermore, Modi (2016) proposed event embedding modeling to conquer the imperfections of the count-based methods. One of the advantages of the event embeddings is enabling to capture the semantic properties of events, it means that the events with different surface form of their constituents but are semantically similar will receive the similar event embeddings. However, one of the disadvantages of these event embeddings is the lack of understanding events with deeper syntactic and semantic analysis of language. Therefore, in this proposed method we present a novel technique to model scripts based on Recursive Neural Networks Semantic Syntactic, namely NESS to represent a sequence of events as composition of their semantic predicates and arguments. The advantage of our proposed model can effectively dealt with sparsity in semantic space by representing meaning at a higher level of abstraction than the surface forms of words.

In the fourth part of Figure 3.1, it presents the way that we enrich our source and target language, which can be considered as an event structure, and then feed into the seq2seq model as the input. In order to do this, we take the output of script modeling then combine with the source and target that we glued from semantic concepts of daily activity location, which detected by a deep neural network. However, this work still has not covered in this thesis due to time limitations.

Chapter 4

Human Activity Recognition from mobile phone sensors

In this chapter, we present our proposed method to deal with imbalanced data for conventional learning methods using a mobile phone sensor. First of all, in Section 4.1 we introduce the problem and related work in Section 4.2. Next, in Section 4.3 we introduce our novel framework, which is using active learning and oversampling technique so that it can solve for the problems of imbalanced data, based on Multilayer Perceptron (MLP) - a machine learning model. Next, the experimental results are illustrated in section 4.4. The chapter ends with a conclusion in Section 4.5.

4.1 Introduction

Human Activity Recognition from wearable sensors and in particular smartphones has been subjected to intense research and industrial activity this last decade Lara and Labrador (2013). Many learning algorithms have been used to classify physical human activities such as *Running, Walking, etc.* as well as interactive and social activities (chatting, talking, playing, etc.). HAR is useful for health monitoring, senior care and personal fitness training as well as for providing context to smartphone applications. Physical human activities are generally classified from recorded sensor data (e.g. accelerometers, GPS, audio, etc.) which are embedded into wearable devices (e.g. smartphones and smartwatches).

HAR systems performances are highly dependent on the classification model (Decision Tree, Support Vector Machine, Multi-Layer Perceptron, etc.), the feature used, the number of classes and the size of the datasets available for training Lara and Labrador (2013). However, another aspect that plays an important role in this domain is the lack of a uniform collection of different activities. In fact, this is the case for most smartphone datasets (e.g. *Running* = 4% and *Walking* =40% distribution). This is called the Class

Imbalance Problem that is known to have a serious influence on the performance of learning algorithms because most standard algorithms expect balanced class distributions He and Garcia (2009).

In the past, research on HAR based on wearable sensors did not systematically handle the class imbalance problem. Therefore, in this work, we introduce a generic framework that integrates active learning with an over-sampling method based on MLP to overcome this problem. We also introduce a new over-sampling method, called BLL SMOTE - an extension of SMOTE Chawla et al. (2002) - which can apply to non-convex spaces.

Contributions. Our contributions are summarized as follows.

- A framework integrating MLP and active learning with over-sampling.
- A new over-sampling method, BLL SMOTE.
- Experiments with 2 available datasets that show the impact of taking the class imbalance problem into account in the learning.

The work is organized as follows. Section 4.2 presents a summary of the state of the art in HAR and in learning techniques with imbalanced data. The overall framework and the BLL SMOTE method are detailed in Section 4.3. Several experiments are reported in Section 4.4. The work ends with a short discussion and an outlook on future work.

4.2 Related Work

Human Activity Recognition (HAR) from wearable sensors data is a very rich domain of research. We restrict here in presenting the main work regarding the classification models being used, the available ecological datasets and the techniques to deal with imbalanced class distribution in data. Regarding the classification models, there have been many approaches to deal with HAR from wearable sensors. Over the last decade, the most common approach is to process windows of data streams to extract a vector of features that will, in turn, be fed to a classifier. Many instance-based classifiers have been used in the field, such as Bayesian Network Lara et al. (2012), Decision Trees Lara et al. (2012); Blachon et al. (2014), Random Forest Blachon et al. (2014), Artificial Neural Network (ANN) [Khan et al. (2010); Kwapisz et al. (2011); Lara et al. (2012)], Support Vector Machines (SVM) [Lara et al. (2012); Anguita et al. (2012)], etc. Since human activities can be seen as a sequence of smaller sub-activities, sequential models such as Conditional Random Fields Blachon et al. (2014), Hidden Markov Model Zhu and Sheng (2009) or Markov Logic Network Chahuara et al. (2016) have also been applied. However, since

the advent of Deep Learning, ANN has become of the most popular model in HAR from wearable sensors [Bayat et al. (2014); Arifoglu and Bouchachia (2017)].

ANN has also been broadly used in HAR [Bayat et al. (2014); Kwapisz et al. (2011); Khan et al. (2010)] and in addition to that deep learning now is a challenging topic that many scientists are interested in HAR, for instance, [Ronao and Cho (2015); Zeng et al. (2014b)] is a demonstration of applying convolution neural networks in HAR using wearable sensors. For applying the machine learning approach in our thesis, we achieved to build a framework of human activity recognition based on a supervised learning algorithm namely *Multilayer Perceptron* as shown in Figure 4.2. It was implemented using Tensorflow library is provided by Google Team. An advantage of this framework allows us to deal with poor data quality by using methods such as *sampling* and *cost-sensitive.* Indeed, the quality of training data is also an extreme challenge, in recent years, a survey He and Garcia (2009) reported that imbalanced data has a serious influence on the performance of learning algorithms. Most standard algorithms expect balanced class distributions, hence datasets contain imbalanced class distribution which makes these algorithms fail to correctly represent the distributive characteristics of the data. As a consequence, it would produce unfavorable accuracies across the classes of the data and leads to a decrease in the total accuracy of learning algorithms. In order to deal with imbalanced data, there are several methods introduced in He and Garcia (2009), namely *sampling, cost-sensitive, kernel-based and active learning*. In the first place, under-sampling (e.g., random under-sampling) was evaluated is as better than over-sampling (e.g., random over-sampling), which means that reduction of the majority classes provides higher accuracy than an increase of minority classes. In addition, the strategy of under-sampling is also important, Yen and Lee (2009) proved that cluster based under-sampling outperforms the other under-sampling techniques. In the second place, cost-sensitive learning methods are concerned with the costs associated with misclassifying examples by using different cost matrices. Several methods based on learning algorithms were introduced such as cost-sensitive data-space weighted adaptive boosting, cost-sensitive decision tree. Although, we only interest in cost-sensitive neural networks in three ways: first, cost-sensitive modification can be applied to the probabilistic estimate; second, the neural network outputs can be made cost-sensitive; third, cost-sensitive modification can be applied to the learning rate. The idea of the probabilistic estimate is integrating cost factors into the testing stage of classification to adaptively modify the probability estimate of neural network output, while the outputs of the neural network are altered during training to bias the neural network to focus more on expensive class. The learning rate can be applied cost-sensitive to put more attention on costly examples during the learning. In the third place, we consider the active learning that has been investigated to compromise with imbalanced learning problems

Ertekin et al. (2007a,b). Generally, active learning is used to solve problems related to unlabeled training data. Moreover, it integrates with sampling techniques to analyze the effect of under-sampling and over-sampling by using uncertainty sampling methodology; the challenge is how to measure the uncertainty of an unlabeled instance in order to choose the maximally uncertain instance to augment the training data. Moreover, this framework also allows for dealing with over-fitting problems such as regularization and dropout.

Machine learning is highly dependent on datasets. It is, even more, the case with Deep Learning. If many papers report work dealing with non-accessible datasets, many others investigate datasets that are made publicly available. The survey by Micucci et al. (2017) presents a large number of publicly available datasets acquired from a smartphone. However, it also shows a lack of uniformity in tasks, sensors, protocol, time windows, etc. It is worth to notice that most of the datasets are restricted to inertial sensors such as accelerometers. The audio sensors are largely ignored while being among the only ones that are always found on a smartphone. It is also worth noticing that some are very imbalanced since the distribution among classes are very different. For instance, in the ExtraSensory Dataset Vaizman et al. (2017), sitting represents 44.2% of the data while running only 0.3%. In this case, the learning approach should consider the class imbalance problem.

He and Garcia (2009) reported that imbalanced data has a serious influence on the performance of learning algorithms, because most standard algorithms expect balanced class distributions. Hence, datasets exhibiting imbalanced class distribution make these algorithms fail to correctly represent the distributive characteristics of the data. As a consequence, it would produce mis-classification of minority classes higher than mis-classification of majority classes, and leads to a decrease in the overall accuracy of learning algorithms. In fact, in HAR, a few studies coped explicitly with this problem such as Abidine and Fergani (2014) who proposed Weighted Support Vector Machines (WSVM) to improve the learning of minority classes. However, the approach is based on a scheme that puts more weight on the errors in the minority classes than on the majority classes. Therefore, this approach is highly dependent on instances of minority classes.

In general, in order to deal with imbalanced data, several other approaches were introduced in He and Garcia (2009) such as over-sampling, active learning. For the former approach, some methods were proposed such as SMOTE Chawla et al. (2002) or Borderline SMOTE Han et al. (2005) which work by generating new synthetic instances of minority classes. Their studies showed that over-sampling techniques succeeded to enhance the classification accuracy for imbalanced datasets. For the latter approach, Ertekin et al. (2007a) introduced a SVM based active learning framework in which SVM starts to train on a given training dataset, then selects the most informative instances from a pool of

training samples, afterward adds the newly selected instances to the training set and finally trains SVM again. This approach has been pursued in the VIRTUAL framework Ertekin (2013). The study showed that active learning can efficiently handle the class imbalance problem. However, all the above-mentioned studies did not combine together in order to settle the imbalanced data problem. Therefore, in this work, we introduce a generic framework to cope with this overall issue. More details will be provided in Section 4.3.

4.3 Oversampling and Active Learning Framework for HAR

Our objective is to improve the learning of HAR model in case of imbalanced datasets. The problem can be defined as follows : Let $A = \{a_1, ..., a_k\}$ be the set of all activities, given a set $T = \{t_1, ..., t_m\}$ of m equally sized time windows, and a set of sensors $S_i = \{S_{i,1}, ..., S_{i,q}\}$. Given a feature space $X \in R^n$, an instance $x \in X$ extracted from sensors S_i at time frame t_j is to be classified, e.g. attached an activity label from A.

In this work, we focus on the classification problem. Our goals are (1) to find a learning algorithm $f : X \rightarrow A$ returning a label $f(x) = a^*$ as close as possible to the actual activity performed during $t_i \in T$, (2) to enhance the classification task using active learning, and (3) to improve the recognition task by over-sampling to balance the imbalanced training set.

In this section, we present the general framework to reach these objectives, and then detail each of its components in the subsequent sections.

4.3.1 Proposed Framework

The framework, as shown in Figure 4.1, is an extension of VIRTUAL Ertekin (2013) to integrate active learning with the oversampling method to overcome the imbalance problem.

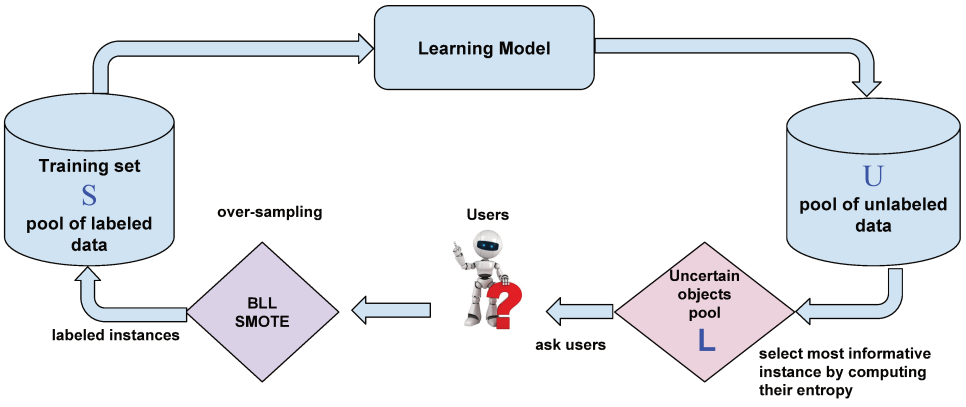


Figure 4.1: The active oversampling framework

First of all, the learning is initiated with a pool of training instances S, which is used to learn a classification model. Then, to choose the most relevant sample to add in the training set, the entropy of each instance from the pool of unlabeled data U is computed using the classifier output by using Equation (4.1). From this, a small pool of uncertain samples L is created by grouping the instances that maximized the Shannon entropy. After that, the small pool L is removed from U and the user is queried for its labels.

Secondly, once L is annotated, our specific over-sampling method, called BLL SMOTE (cf. Section 4.3.4), looks for minority instances inside the pool L and generates new artificial instances of these minority classes. The original pool L plus the generated instances of the minority classes are added to the training set S and the training restarts. This means that at each iteration, the training set is bigger but less and less imbalanced. Each part of this framework is detailed in the following sections.

4.3.2 Classification Model: Multi-Layer Perceptron

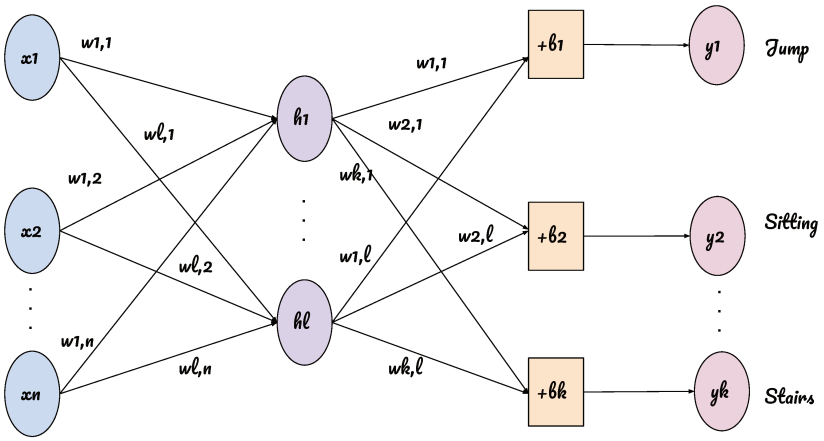


Figure 4.2: Multilayer Perceptron

For activity classification, many existing techniques such as SVM Anguita et al. (2012) or Random Forest Blachon et al. (2014) can be used. Among them, MLP is one of the most common methods used in HAR systems. In the rest of this work, MLP is used as a classification model. This choice is, on the one hand, justified by the fact that it demonstrates high performances on the task as well as high promises of improvement and, on the second hand, by the incremental learning strategy that fits well with active learning.

MLP can be seen as a class of feed-forward neural network composed of at least three layers of nodes, namely input, hidden and output layer. MLP can be learned using the back-propagation method, an efficient optimization method that operates iteratively. More precisely, our MLP network is designed as follows. Given inputs $X = \{x_1, ..., x_n\}$ are

the features extracted from the sensors, the hidden layer nodes $H = \{h_1, ..., h_l\}$ and the output layer nodes $Y = \{y_1, ..., y_k\}$ are computed as follows:

$$\begin{bmatrix} h_1 \\ . \\ . \\ . \\ h_l \end{bmatrix} = \begin{bmatrix} w_{1,1} * x_1 & +... & +w_{1,n} * x_n \\ . & . & . \\ . & . & . \\ . & . & . \\ w_{l,1} * x_1 & +... & +w_{l,n} * x_n \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ . \\ . \\ . \\ y_k \end{bmatrix} = \begin{bmatrix} w_{1,1} * h_1 & +... & +w_{1,l} * h_l \\ . & . & . \\ . & . & . \\ . & . & . \\ w_{k,1} * h_1 & +... & +w_{k,l} * h_l \end{bmatrix} + \begin{bmatrix} b_1 \\ . \\ . \\ . \\ b_k \end{bmatrix}$$

This is shown in Figure 4.2, where $\{w_{1,1}, ..., w_{l,n}\}$, $\{w_{1,1}, ..., w_{k,l}\}$ and $\{b_1, ..., b_k\}$ represent weights and bias.

4.3.3 Active Learning

The principle of Active Learning (AL) is to learn to label unknown instances by selecting (querying) some specific instances and ask an external system (e.g., a human operator) to label them. It has become an emerging research topic with applications in many fields such as image segmentation Biswas and Jacobs (2012), data clustering Mai et al. (2016) text classification Tong and Koller (2001). Applying AL in HAR is thus an interesting approach since it can further boost up the accuracy by involving humans in the classification task, especially for hard to classify activities. Moreover, its scheme provides a natural way to cope with data imbalance by exploring some most uncertain data spaces, as pointed out in Ertekin et al. (2007a).

Typically, an active learning algorithm chooses objects that their labels are among the most uncertain ones to query users for. Uncertain instances can be chosen in many different ways Settles (2010). Our technique is built upon the uncertainty sampling technique Settles (2010) whose principle is that the most relevant instances to be selected for annotation are the ones for which the estimates are the less certain. Thus, after MLP training, we predict the labels of U using the training output Y of MLP. Y can be seen as a vector of the probability of labels. Then the instances in U are ranked according to their decreasing Shannon Entropy, because the higher entropy of an instance is, the more uncertainty there is on its class. Therefore, the most uncertainty instance can be picked up by maximized Shannon entropy Shannon (2001) using Equation (4.1):

$$x_H^* = arg\max_x - \sum_i P_\theta(y_i|x) \log P_\theta(y_i|x) \tag{4.1}$$

where x is an instance, $P_\theta(y_i|x)$ is the probability of all possible labels on the instance.

4.3.4 Oversampling Border Limited Link SMOTE Method

While classical active learning methods only add in the training set the uncertainty instances, that were labeled by a user, our method also performs over-sampling on queried data. This makes it possible to put new information into the training set and tackle the class imbalance problem.

Over-sampling consists of adding a new sample to a training set, whether they are synthetic or real. For instance, SMOTE Chawla et al. (2002) generates a new synthetic instance, using Equation (4.2):

$$x_{new} = x_i + (x_i^\theta - x_i) * \lambda \tag{4.2}$$

where x_{new} is the new sample generated from $x_i \in S_{min}$, with S_{min} is the samples of minority class, x_i^θ is one of the k-nearest neighbors of x_i: $x_i^\theta \in S_{min}$, and $\lambda \in [0, 1]$ is the random number, which allows to randomly generate the new synthetic instance x_{new} along the line between x_i and x_i^θ.

However, this method is not relevant in the case of non-convex spaces. For instance, imagine space as represented in Figure 4.3. If x_i and x_i^θ are two samples of the green (circle) class, a direct application of Eq. (4.2) would produce a new sample x_{new} which would not be in the right space.

To avoid the mis-generation of synthetic instances in the case of non-convex dataset, we introduce the BLL SMOTE method described as follows. The method uses Eq. (4.2) but calculates the distance from x_{new} to each of the k-nearest neighbors of x_i, denoted as $d_j = d(x_{new}, x_i^\theta),\ j = 1, ..., k$, where d is the Euclidean distance. Then, the distance of the artificial instance x_{new} with its nearest instance $x_{diff} \notin S_{min}$ such that $x_{diff} \in S$, denoted as $d_{diff} = d(x_{new}, x_{diff})$ is computed. Finally, each d_j is compared to d_{diff}. If any d_j is greater than d_{diff}, then this artificial instance x_{new} is not accepted to be generated. Otherwise, x_{new} is accepted.

An advantage of BLL SMOTE is to avoid the mis-generated new synthetic instance in non-convex datasets

4.3.5 Implemented Framework

The overall learning process is summarized in Algorithm 1. Providing a training set S, a pool of instances U and a number of query N, an MLP model Ω is firstly trained from S (*line 4*). Next, uncertainty sampling is used to query instances to be added to S (*line*

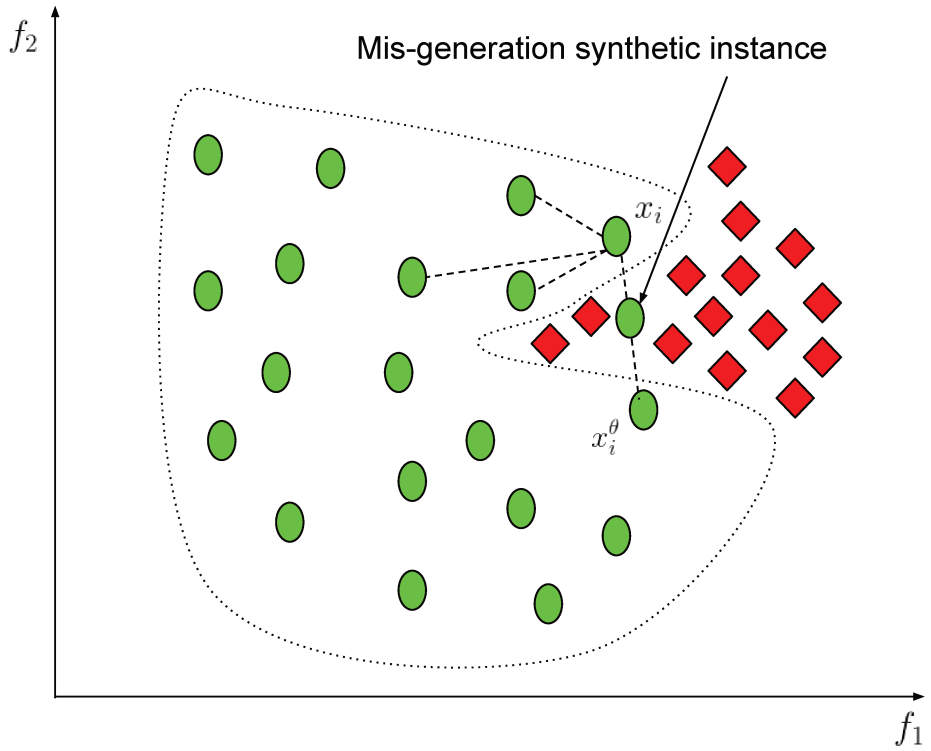


Figure 4.3: Example of mis-generation synthetic instance in non-convex dataset

5). MLP matrix output is then used to calculate the entropy of each instance from U by using Equation 4.1. The instances are stored in the small pool of uncertainty sampling L which is removed from U (*line 6*). Once L is extracted, labels of each instance of L are retrieved (*line 7*). S is then added to the training set (*line 8*).

Second, since the small pool of uncertain objects L is produced, our oversampling BLL SMOTE (cf. 4.3.4) method looks for minority instances inside the pool L and generate new artificial instances of these minority classes. For each instance x_* in L, the neighbors are collected (*line 10*). For each of these neighbors x_*^θ a new sample is created and inserted to the training set if not too close to an instance of another class (*line 11-21*).

4.4 Experimental Analysis

4.4.1 Dataset

To perform HAR, we restricted ourselves to comparable datasets that contain at least audio data and accelerometer data which are the only sensors that are guaranteed to be found on any smartphone. We selected the LIG SmartPhone Human Activity Dataset (LIG-SPHAD) Blachon et al. (2014) and the ExtraSensory Dataset Vaizman et al. (2017) which are both publicly available, do contain continuous audio and accelerometer data and are annotated using physical human activity labels. The LIG-SPHAD has been acquired on

Algorithm 1: Algorithm MLP AL OS Border Limited Link SMOTE

Input:

(a) $X = \{x_1, ..., x_n\}$: training instances

(b) S: pool of training instances for MLP

(c) U: pool of unlabeled instances

(d) N: number of query

Output:

Ω : the learned model

Initialize S from a subset of X
Initialize $L \leftarrow \emptyset$
for $i : 1 \rightarrow N$ **do**
    $\Omega = MLP\_Train(S)$
    $L \leftarrow Uncertainty\_Sampling(U, \Omega)$
    $U \leftarrow U - L$
    Query labels y_* for all $x_* \in L$
    $S \leftarrow S \cup L$
    foreach $x_* \in L$ **do**
        $L\_neighbors$ = k_nearest_neighbors (x_*)
        foreach $x_*^\theta \in L\_neighbors$ **do**
            $x_{new} = x_* + (x_*^\theta - x_*) * \lambda$
            $d_{diff} = Euclidean\_distance(x_{new}, x_{diff})$
            $accepted\_generation = True$
            foreach $x_*^\theta \in L\_neighbors$ **do**
                $d = Euclidean\_distance(x_{new}, x_*^\theta)$
                if $d > d_{diff}$ **then**
                    $accepted\_generation = False$
                    break;
            if $accepted\_generation = True$ **then**
                $S = S \cup (x_{new}, y_*)$
return Ω

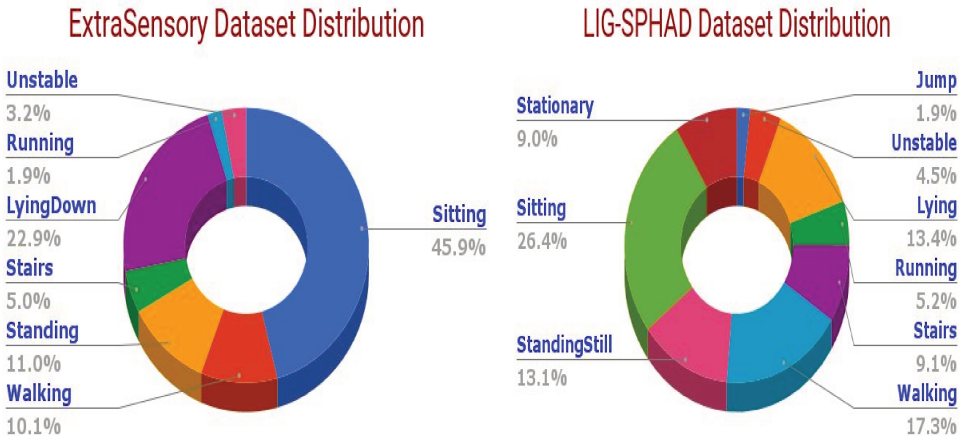


Figure 4.4: Distribution of the activity labels over the datasets

19 different subjects (14 males, 5 females) within an age bracket of 19-29 years old. Each subject performed 9 different activities (walking, running, jumping, unstable, stairs, lying, standing still, stationary, sitting) using 4 smartphones placed on different usual places (pocket, hand, bag). Accelerometer and audio data were captured using the RecordMe application Blachon et al. (2014). The sensor signals were sampled in a fixed-width sliding window of 2 seconds and %50 overlapping. 12850 instances are available in the dataset that we split into a training set (9637 instances) and a test set (3213 instances). The ExtraSensory Dataset has been acquired on 60 users that were engaged in their regular daily life activities in the age range of 18-42 years old. Several sensors such as accelerometers, gyroscope, audio, etc. were recorded and seven main physical activities were self-reported: *lying down, sitting, standing in place, standing and moving, walking, running, bicycling* plus 109 labels describing more specific context. Features were computed over a 20-second long buffer without overlapping. For the purpose of this study we extracted only 6 activities *Walking, Running, Stairs, Sitting, Lying-down, Standing* that were double-checked by the experimenters and merged the other activities in the *Unstable* class. In the end, a total of 13489 instances was used that we randomly split into a training set (10117 instances) and test set (3372 instances). As can be seen from Figure 4.4, the two datasets are imbalanced. For instance, in the LIG-SPHAD, *jump* is the smallest distribution class (1.9%) compared to the highest one *sitting* (26.4%). For the ExtraSensory Dataset, *running* (1.9%) is a minority class while *walking* (45.91%) is by far the most frequent one. The datasets were randomly split into a training set and test set for classification task.

4.4.2 Baseline results with the MLP

The MLP we implemented is composed of three layers as described in Section 4.3.2. The TensorFlow library was used to implement MLP. The experiment conducted on a

workstation with 3.2Ghz CPU and 16GB RAM.

The learning results are presented in Figures 4.5 for the two datasets. The blue line corresponds to the F1 score on the test set. On LIG-SPHAD the overall score is 68% F1 while it is about 65% on the ExtraSensory Dataset. At the beginning of the learning phase, the F1 score of minority classes are very low while the F1 score of majority classes are high. At the end of learning, the F1 of majority classes still have a high score while the F1 score of minority classes steadily increase but stay below the overall score. It shows that, as every discriminative learning that does not naturally take the imbalanced class problem into account, the learning favors majority classes.

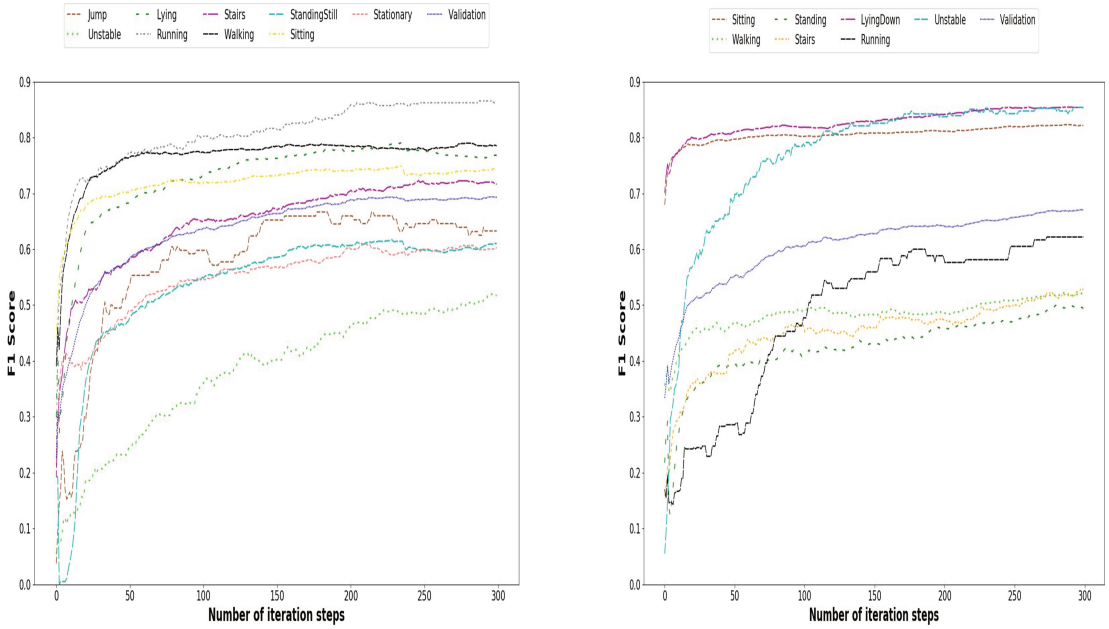


Figure 4.5: Baseline learning curve of the MLP on LIG-SPHAD (left) and ExtraSensory Dataset (right).

4.4.3 MLP learning with BLL SMOTE

The MLP was then learned using the BLL SMOTE method. In this experiment, BLL SMOTE is parametrized using a query budget limitation σ of 950, a query size of $\alpha = 50$ and a neighborhood size k of 6. Different learning tasks were carried out using either: (i) A random AL: the instances are picked up randomly from U and added in S; (ii) AL without over-sampling: only the most uncertain instances with the largest entropy are chosen and added to S; (iii) AL with SMOTE: the most uncertain instances are chosen, then new instances of the minority classes are created without taking (non-)convexity of the instances space into account. Then they are added to S; (iv) AL with BLL SMOTE: our method. Unless specified otherwise, the parameter values are the same for all methods.

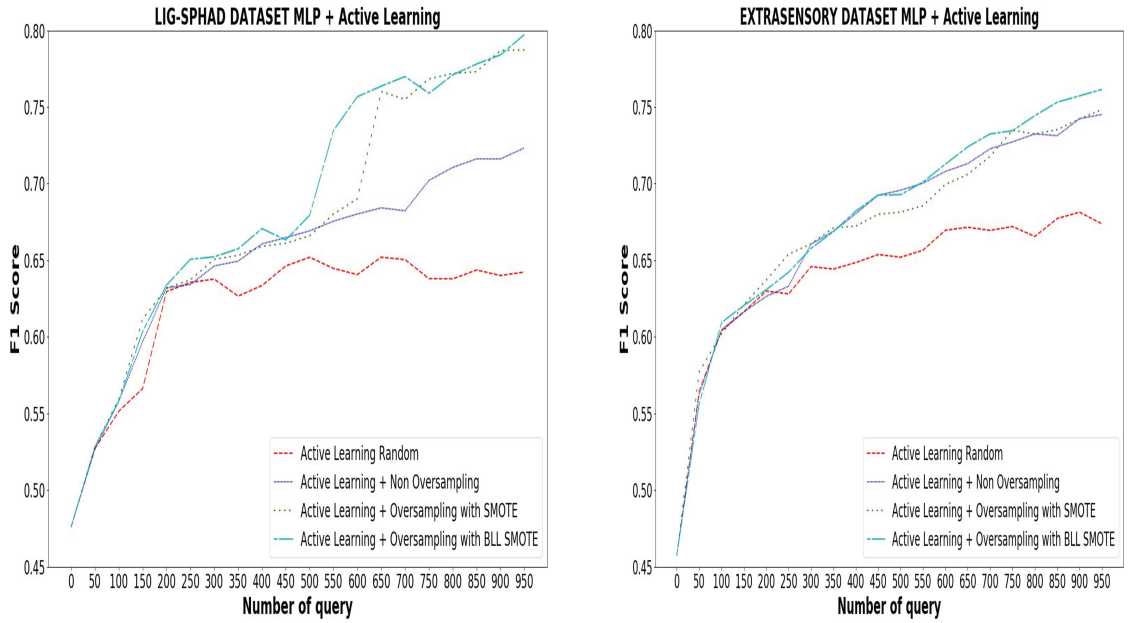


Figure 4.6: MLP + Active Learning on LIG-SPHAD (left) and ExtraSensory Dataset (right).

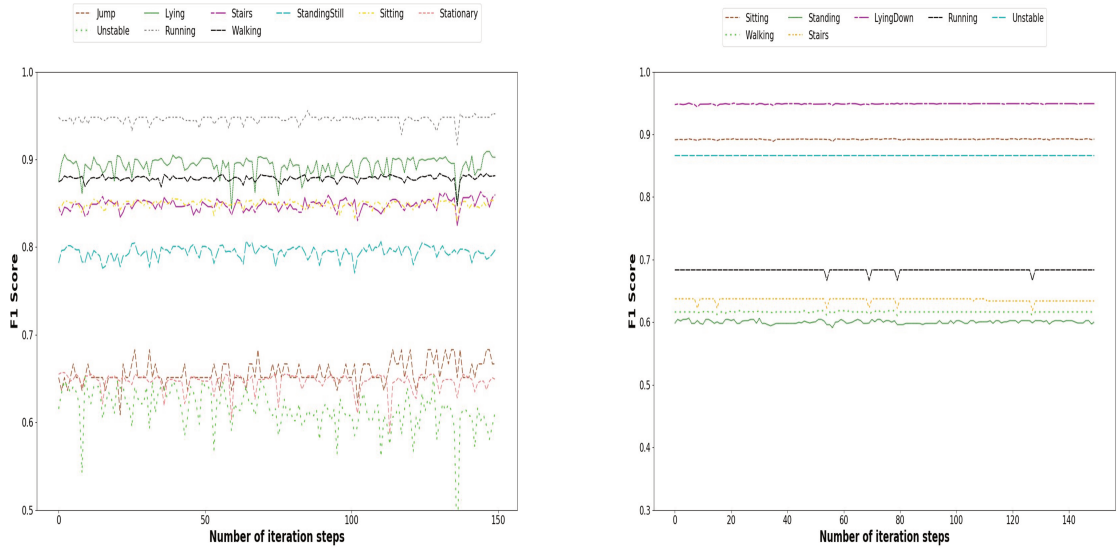


Figure 4.7: Last step of Multilayer Perceptron after last query of Active Learning and Over-sampling BLL SMOTE on LIG-SPHAD (left) and ExtraSensory Dataset (right)

Figure 4.6 shows the F1 score curves of the four methods on the test set of the two datasets. For LIG-SPHAD on the left side of Figure 4.6, BLL SMOTE gave the best performances reaching 80% far better than the original 68%. For the ExtraSensory dataset, BLL SMOTE also gave the best result reaching 76%, which is better than the previous performance of 65%. However, the difference w.r.t the other methods is less pronounced. In any case, these results show that AL and over-sampling greatly improve global performances in case of imbalanced data.

BLL SMOTE also has an effect on the classification performance of each class. On Figure 4.7, the MLP performance on LIG-SPHAD at the last step of active learning with over-sampling BLL SMOTE, demonstrates that the minority classes such as *Jump, Unstable* can achieve nearly 0.7 and 0.65 F1 score respectively, that is much higher than the MLP performance in Figure 4.5 where F1 score are 0.6 and 0.5 respectively. On the ExtraSensory dataset, the right side of Figure 4.7 also illustrates that minority classes such as *Running, Stairs* can reach an F1 score of 69% and 65% respectively higher than in the right side of Figure 4.5, where the same classes achieved 60% and 50% F1 score respectively. Hence, BLL SMOTE makes it possible to increase minority class performance in a discriminative setting.

4.5 Conclusion

In this work, we introduced a generic framework that integrates active learning with the over-sampling method based on MLP to overcome the class imbalance problem. We also introduce a new over-sampling method, called BLL SMOTE - an extension of SMOTE Chawla et al. (2002) - which can be applied to non-convex spaces.

The experiments carried out on two different datasets demonstrated that using active learning with over-sampling to tackle the imbalance distribution of class can increase the global F1 score of the two datasets by about 15% absolute over the baselines. In each case, BLL SMOTE shows slightly higher performances than using SMOTE plus Active learning. In addition, BLL SMOTE is able to increase the classification performance of minority classes. Another important point of this study is the fact that our method prevents the mis-generation of the synthetic samples, thanks to its capacity to manage non-convex datasets.

These results show two advantages over classical approaches: the method makes it possible to improve overall and local performances and does not require extra external data. This last advantage is important in a domain such as a smartphone HAR where data collection is costly and available datasets might differ too much in terms of target, features or time resolution.

Chapter 5

Scripts Generation from Events

In this chapter, we present our novel approach to generate scripts from events, which are detected using a deep learning method on wearable sensors data. Section 5.1 shows an introduction, then related works are presented in section 5.2. In section 5.3 we introduce our proposed methods, and the experiment results are illustrated in section 5.4. The chapter ends with a short discussion and an outlook of future work in Section 5.5.

5.1 Introduction

Script is a stereotypical sequence of events in prototypical scenarios, was first introduced by Schank and Abelson (1975). As mentioned in Schank and Abelson (1975) a story has invoked a script with one or more interesting deviations, and more precisely script is also a very boring little story. There are extremely numerous scripts in our daily life such as birthday party script, working day script, and restaurant script, and so on. For instance, Table 5.1 shows an example of restaurant scripts in different scenarios.

The above scripts enable us an insight into the sequence of events happened in the restaurant context with different scenarios, which are providing in the form of human readable texts. For instance, the sequence of events in scenario 2 let us deduce that *the person is ordering food*, or scenario 4 shows us that *the person is leaving the restaurant*. In general, scripts allow human to deeply understand everyday human life in the real world.

Fortunately, nowadays HAR systems are using wearable sensor data also permit us to recognize human activities during daily life, which are exploiting machine learning models [Anguita et al. (2013); Kwapisz et al. (2011); Blachon et al. (2014)] to classify and predict human activity such as *walking, running, sitting, etc.*. However, these current HAR systems are often tackling by recognizing discrete daily human activities, hence it is not sufficient to constitute a script. In this work, we have therefore proposed a novel approach in order to automatically generate scripts from the human activity recognition

Scenario 1: Entering
Self into restaurant.
Look where empty tables are.
Where to sit.
Self to table.
Sit down.
Scenario 2: Ordering
Receive menu.
Read menu.
Decide what self want.
Order to waitress.
Scenario 3: Eating
Receive food.
Eat food.
Scenario 4: Exiting
Ask for check.
Receive check.
Tip to waitress.
Self to cashier.
Money to cashier.
Self out of restaurant.

Table 5.1: Event sequence of restaurant visiting

system using wearable sensor data. Figure 5.1 depicts a process of the raw signal data (e.g., accelerometers, audio, etc.) of human activity into the generation of the scripts. First, our method generates a rich semantic concept (e.g., gesture, activity, location) of raw signal sensor data (e.g., accelerometers) detected by using the Sequential HAR system, which will be described in section 5.3. Unlike Miyanishi et al. (2018) semantic concepts are unknown related to each other's, therefore Miyanishi et al. (2018) have to develop temporal interactions among semantic concepts to remove objects and places unrelated to each action. In contrast, our sequential HAR system allows us to predict semantic concepts over time, and present it in the form of a sequence. And second, we propose to formulate the generation of scripts as a hybrid combination of semantic concepts detection templates and a machine translation problem. In more detail, detection rules provide several templates to combine semantic concepts and generate sentences as a target language for machine translation problems, which is using semantic concepts as source language and the generated sentences as the target language, will be detailed in

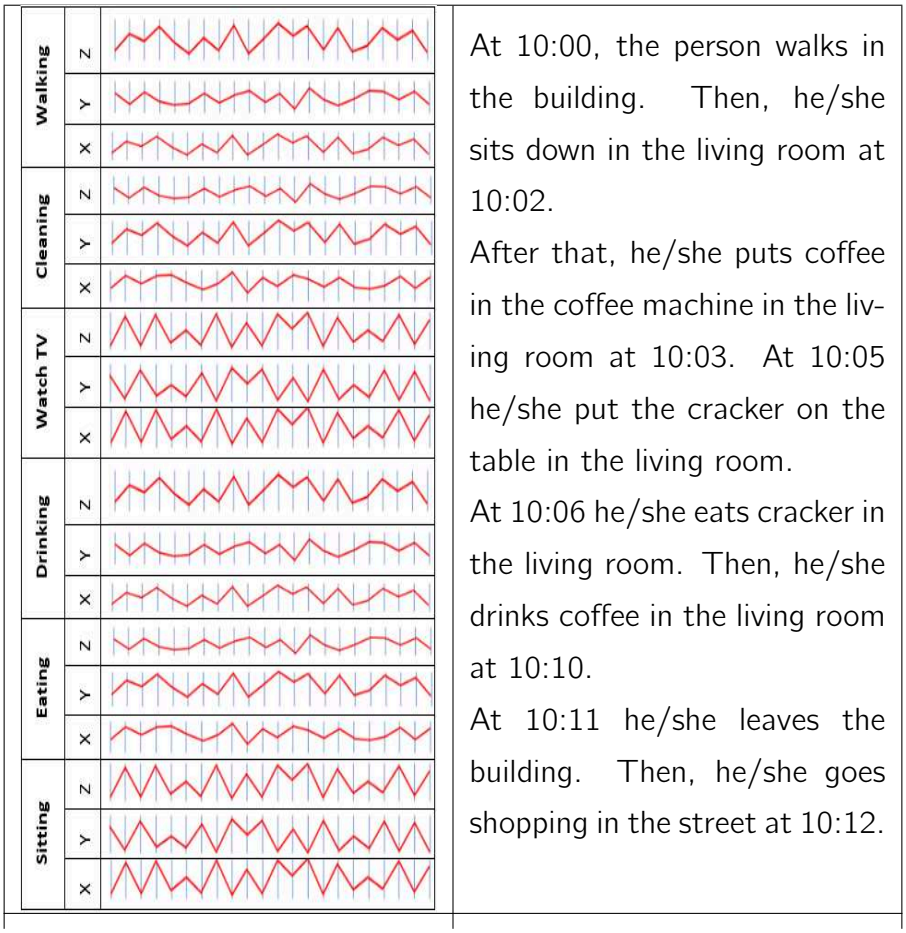


Figure 5.1: Illustration of generating scripts from human activity using wearable sensor data.

section 5.3.

We evaluate our proposed method on a daily-activity dataset collected by several subjects on several days Sztyler et al. (2016). The experimental results present that the proposed method can automatically generate scripts in unseen locations with high relevant accuracy corresponding to the actual texts. Furthermore, we found that our method is the first approach that enables us to generate automatically scripts from the HAR system using wearable sensor data.

The remainder of the work is organized as follows. Section 5.2 presents a summary of the state of the art in text generation from the HAR system. The proposed methods are detailed in Section 5.3. Several experiments are reported in Section 5.4. The work ends with a short discussion and an outlook of future work in Section 5.5.

5.2 Related Work

In recent years, there have been some researches interested in activity report generation in health care [Kashimoto et al. (2017); Inoue et al. (2015)]. However, these existing studies

focus on generate activity reports based on charts or graphs by predicting activity labels from sensor data. In contrast, in this work, we mainly focus on generating a *sequence of events/scripts*.

For recognize human activities of daily life, many approaches [Anguita et al. (2013); Kwapisz et al. (2011); Bayat et al. (2014)] were proposed used online and offline classification models, which allow recognizing indoor and outdoor activities with external and wearable sensors. Moreover, due to the evolutionary development of wearable sensors, it can be easy to integrate on wearable devices (e.g., smart-phone, smartwatch, etc.) to record attributes (e.g., acceleration, location, etc.). Recently, many vision-based activity recognition methods have been proposed using wearable cameras [Ma et al. (2016); Pirsiavash and Ramanan (2012); Ohnishi et al. (2016)] for predicting daily activities and their related objects in home environments. With this motivation in our current work, we are also using wearable sensors to predict daily activities by developing a deep learning model. However, we still have not considered the wearable camera for our HAR system. For context-aware of human activities of daily life, some studies were carried out [Lee and Mase (2002); Riboni and Bettini (2011)] to recognize daily activities, and simultaneously the places where the activities were performed. According to Lee and Mase (2002), the transition of locations is detected by integrating the subject's motor activities using a dead-reckoning method. Similarly, Riboni and Bettini (2011) proposed a system that combines between ontological reasoning and statistical inferencing to recognize more accurately activities based on contextual conditions (e.g., location, surrounding environments, used objects). Nevertheless, these researches did not consider on the sequence of events or scripts generation from daily activities recognition.

There have been much extensive researches work on language description generation [Miyanishi et al. (2018); Rohrbach et al. (2013)] from visual content (e.g., image, video, etc.). In general, there are four main research directions according to Rohrbach et al. (2013) (1) generating descriptions for images and videos which already contain some associated texts [Aker and Gaizauskas (2010); Feng and Lapata (2010)]; (2) generating descriptions with manually defined rules or templates [Tan et al. (2011); Guadarrama et al. (2013)]; (3) retrieving existing descriptions from similar visual content [Farhadi et al. (2010); Ordonez et al. (2011)]; (4) learning a language model from training corpus to generate descriptions [Kulkarni et al. (2011); Kuznetsova et al. (2012)]. However, these current work did not focus on description generation from wearable sensor data (e.g., accelerometers, audio, etc.), which is much more limited information than images or videos.

5.3 Proposed Method

In order to automatically generate scripts, we design the proposed system illustrated in Figure 5.2. In the proposed system, multiple semantic concepts were produced from sensor data based on learning Sequential Deep Learning Model so that it can classify and predict three real-world properties (gesture, activity, location) be considered as semantic concepts and events. We will detail these processes in the next section 5.3.1.

5.3.1 Semantic Concepts Generation with Deep Learning Neural Network Model

Semantic Concepts Representation and Events

In the first step, we represent the real-world states as a sequence of events $e = [e_1, e_2, ...e_n]$, where each event e is denoted as a tuple of semantic concepts $(sc_1, sc_2, ..., sc_m)$. Each semantic concepts is represented by concept form of $< g, a, l >$, where g is the gesture of the subject (i.e., Walking, Running, Sitting, etc.), a is the activity of the subject does (i.e., Mealpreparation, Deskwork, Housework, Shopping, etc.), l is the location (i.e., Street, Home, Building, etc.) the subject stands.

Generating Semantic Concepts using Deep Neural Network Model

In this section, we present the deep learning Neural Network Model with a sequence-labeling approach that is used to generate semantic concepts of gesture, activity, and location by predicting labels from sensor data as illustrated in Figure 5.2.

First, we process a time-sliced and multi-dimensional sensor data on a 3-axis signal of accelerometer data that was collected by a mobile device carried around several persons' body-positions (e.g., Thigh, Upper-arms, Head, Waist, Forearms, Shin, Chest). Actually in HAR problem, most of recent studies [Kwapisz et al. (2011); Anguita et al. (2012); Khan et al. (2014)] used a fixed sliding window for features extraction and built different machine learning models (e.g., Random Forest, Support Vector Machine, Convolutional Neural Network, and so on) to classify and predict the human activity labels. However, these current works relied on heuristic hand-crafted feature design, which can not achieve to find the best features for the learning model to accurately classify different human activities. Therefore, in this work we follow Jiang and Yin (2015a) to build a deep neural network – Deep Neural Network (DNN) captures the salience of signal (3-axis accelerometer data) in different scales by a time-sliced representation as shown in Figure 5.2. For the DNN model, assume that we have a signal of sensor data in duration $T = (t_1, t_2, ..., t_N)$. At

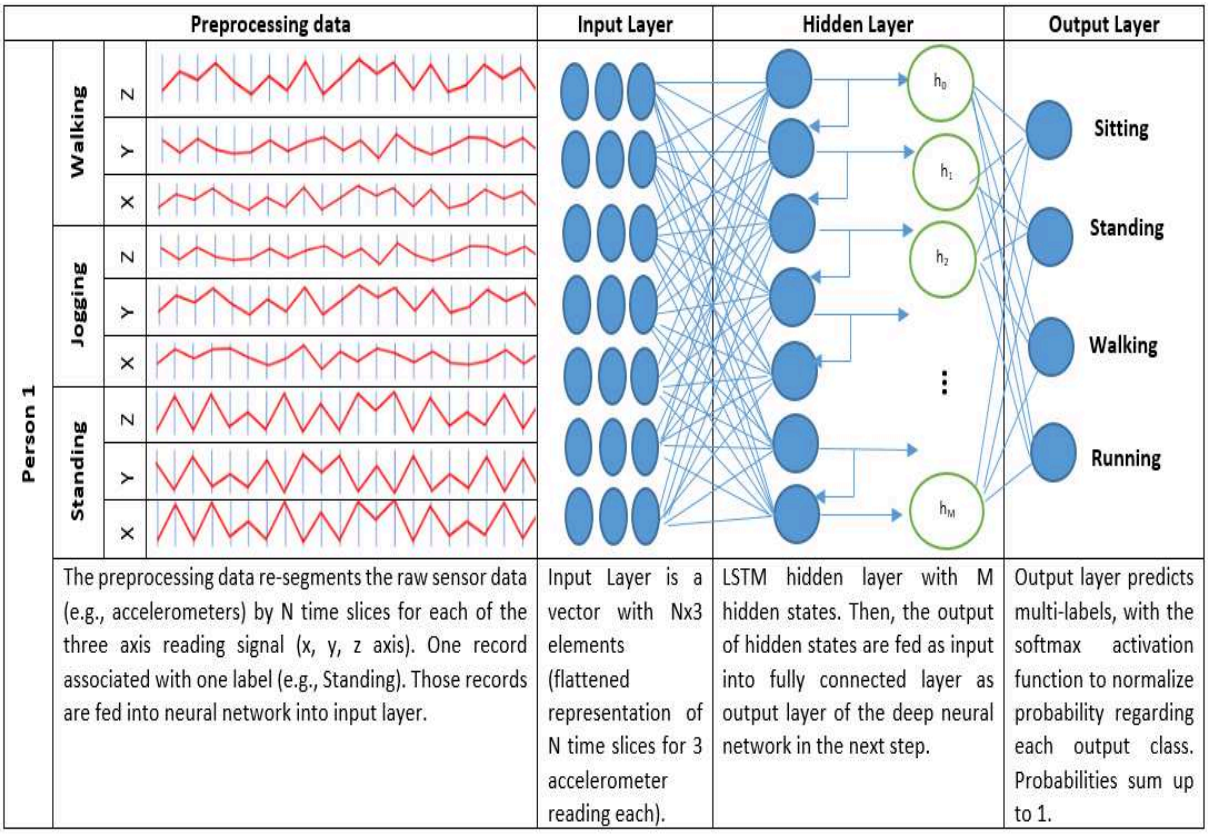


Figure 5.2: Illustration of generating semantic concepts (e.g., gesture) from human activity using deep neural network.

each step t is defined by a time-sliced that segments raw signal from start time t_i to end time t_j has an input vector $x_{t_{ij}}$ ($x_{t_{ij}}$ is composed by 3-axis signal of sensor data), and a label of sensor data $y_{t_{ij}}$ (one-hot vector) is the most frequent label that is represented in the time duration t_i to t_j, and produce a hidden state $h_{t_{ij}}$. The function of LSTM is defined as below:

$$g_{t_{ij}} = sigmoid(W_g x_{t_{ij}} + U_g h_{t_{ij}-1} + b_g)$$
$$s_{t_{ij}} = tanh(W_s x_{t_{ij}} + U_s h_{t_{ij}-1} + b_s)$$
$$f_{t_{ij}} = sigmoid(W_f x_{t_{ij}} + U_f h_{t_{ij}-1} + b_f)$$
$$\hat{s_{t_{ij}}} = f_{t_{ij}} * s_{t_{ij}-1} + g_{t_{ij}} * s_{t_{ij}}$$
$$o_{t_{ij}} = sigmoid(W_o x_{t_{ij}} + U_o h_{t_{ij}-1} + b_o)$$
$$h_{t_{ij}} = o_{t_{ij}} * relu(\hat{s_{t_{ij}}})$$

where $W_g, W_s, W_f, W_o \in \mathbb{R}^{n_H X n_I}$ and $U_g, U_s, U_f, U_o \in \mathbb{R}^{n_H X n_H}$ are weighted matrices. The dimension n_I is the size of the input vector, n_H is the size of hidden vector. b_g, b_s, b_f, b_o are bias vectors, $*$ is the element-wise multiplication. $g_{t_{ij}}, s_{t_{ij}}, f_{t_{ij}}, \hat{s_{t_{ij}}}, o_{t_{ij}}$ are the input gate, states of memory cells, forget gates of the memory cells, new states of memory cells and output gates of memory cells, respectively. We refer to the above function as $h_{t_{ij}} =$

$LSTM(x_{t_{ij}}, h_{t_{ij}-1})$. Then, the output of hidden states $h_{t_{ij}}$ are fully connected to an output layer with activation softmax that produces the probability of all possible labels. Next, we predict labels of sensor data $x_{t_{ij}}$ at each time-sliced window t_{ij} by maximizing the conditional probability $p(y_{t_{ij}}|x_{t_{ij}}) = \frac{\exp(W_o h_{t_{ij}} + b_o)}{\sum_k (W_o h_k + b_o)}$, where $W_o \in \mathbb{R}^{n_L X n_H}$. The dimension n_L are the size of labels. Afterward, we combine simultaneously labels into semantic concepts with its start and end times. Consequently, we successfully capture a chronological semantic concept that composes of three conceptual components gesture, activity, the location from sensor data.

5.3.2 Scripts Generation from Semantic Concepts

In this section, we will introduce the technique to generate scripts from the semantic concepts that we detected by using deep learning neural network model, which is described in section 5.3.1. First, we will generate *target sentences/sentence descriptions* from the semantic concepts by manually defined detection rules. Second, we will enrich the semantic and syntactic of the target sentence by learning sequence to sequence model.

Scripts Generation from Manually Detection Templates

In recent years, manually defined rules and templates were used in several studies [Tan et al. (2011); Kulkarni et al. (2011); Kuznetsova et al. (2012)], which allowed to generate language from extracted semantic concepts from visual content (i.e., images, videos). In this section, we follow this technique in order to generate *target language/sentences* from the semantic concepts, which are extracted from outputs of recognition by using deep learning neural networks as presented in the section 5.3.1. To generate target sentences/ language, we consider a triplet of concepts: human gesture concept (e.g., walking), human action concept (e.g., meal preparation) and location concept (e.g., home) the places where the action was done. Simple rule-based methods rely on these concepts. Our first template is based on the subject who is known as first-person wore the wearable sensors, therefore to be more generic the subject phrase is "the person". Then, it is concatenated with the human gesture, action and location concepts according to identified gesture/action/location concepts. For instance, if *"sitting"* and *"meal preparation"* are detected simultaneously, then we form a sentence likes *"the person sits and prepares a meal"*. If the location *"home"* is identified, then we output *"the person sits and prepares the meal at home"*. Sometimes, the location concepts might be identified but gesture and action concepts can be missing. In this case, we use a phrase to present location setting only such as *"the person is at home"*. The figure 5.3 presents the simple rules for the detection templates. As a result, the target sentences/language was created by using

these simple templates. However, templates are able to provide clues indicating what is happening in wearable sensor data but it is still far from the perfect, and also as mentioned in Kuznetsova et al. (2012) using templates limits the natural flexibility of language. For this reason, in the next section, we will apply the sequence to sequence model in order to learn a language model from a training corpus to automatic generate efficiently scripts.

```
IF GESTURE && ACTION && LOCATION != "unknown"
        => The person GESTURE + ACTION + LOCATION
ELSE
        IF GESTURE
        => The person GESTURE
        ELSE IF ACTION
        => The person ACTION
        ELSE IF LOCATION
        => The person LOCATION
```

Figure 5.3: Illustration of scripts generation from manually detection templates.

Scripts Generation from Sequence to Sequence Model

In this section, we will introduce the method to automatically generate scripts by using the sequence to sequence model (seq2seq) Sutskever et al. (2014b). In fact, seq2seq model is widely used in Natural Language Generation (NLG), there have been many researches [Rohrbach et al. (2013); Guadarrama et al. (2013); Venugopalan et al. (2015b,a, 2016)] work on captions generation for videos and images. However, in order to approach our problem (for accelerometers sensors data description), we propose to formulate the generation of natural language as a machine translation problem by applying seq2seq model, which contains two Recurrent Neural Networks (RNNs) named as encoder and decoder. The encoder reads input is the sequence of semantic concepts, which are extracted from the recognition outputs of the DNN model, called as the source for the seq2seq model $(sc_1, sc_2, ..., sc_m)$, then encoder RNNs produces an output as a single vector. After that, the decoder is another RNNs reads the last encoder output vector (also called context vector c) with given sequence of target $(s_1, s_2..., sm')$, which are generated from the manually detection templates as presented in last section, to produce an output sequence of

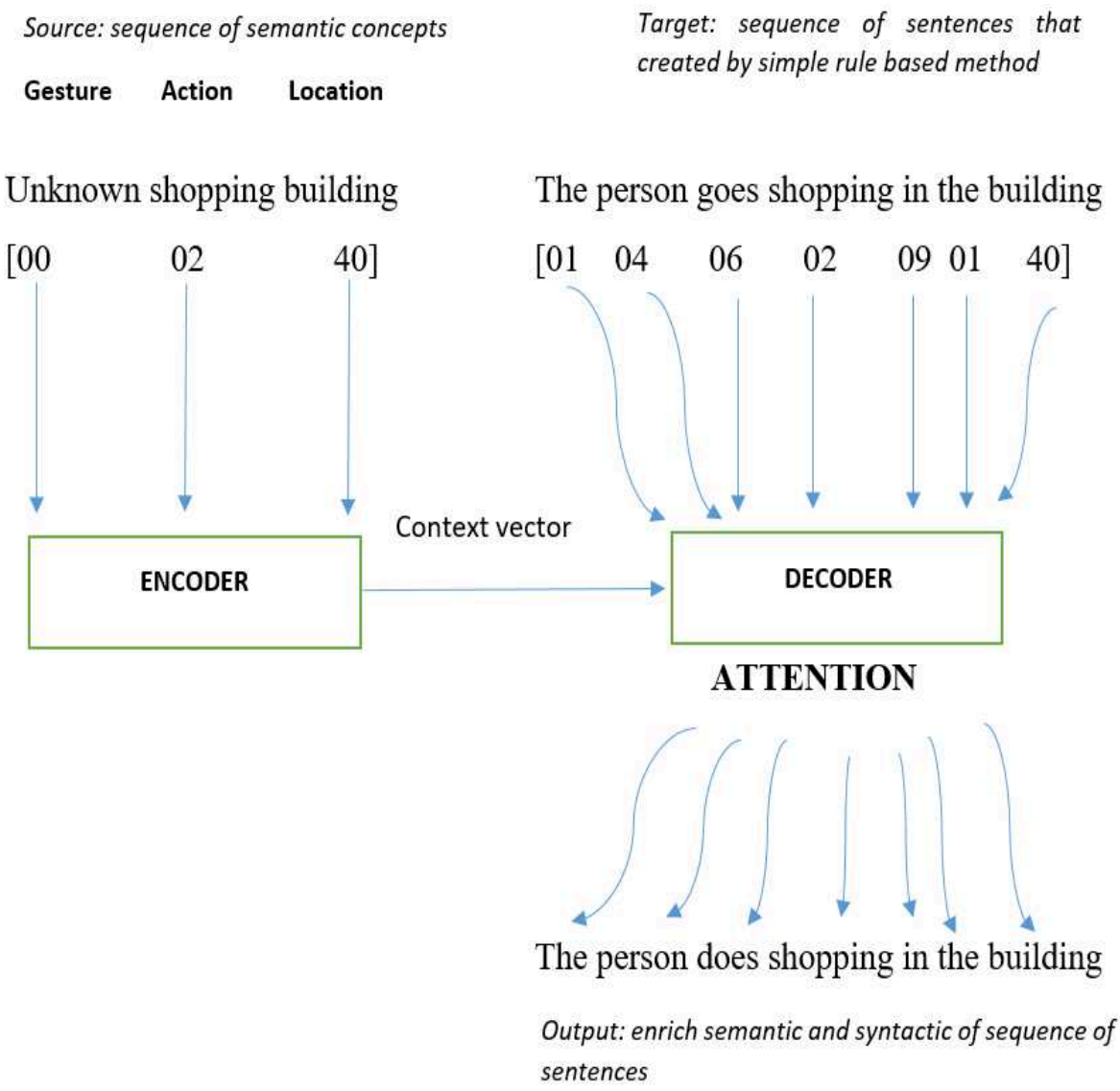


Figure 5.4: Illustration of seq2seq model for enrich semantic and syntactic of scripts generation from semantic concepts.

target sentences named as $(s_1, s_2..., sm)$. The function of RNNs defined as follow:

$$h_t = \delta_h(W_h x_t + U_h h_{t-1} + b_h)$$
$$y_t = \delta_y(W_y h_t + b_y)$$

However, with the simple decoder seq2seq might not be efficient to generate corrected words for the sequence of words. Therefore, in this seq2seq model, we also apply attention decoder [Qader et al. (2018); Britz et al. (2017)] by multiplied attention weights with encoder output vectors to produce the weighted combination. This result obtains a specific part of the input sequence; therefore, decoder attention can choose the right words to generate. We follow Qader et al. (2018) to produce attention weights in decoder by calculating the following Equation 5.1.

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})} \tag{5.1}$$

where e_{ij} is computed as follow: $e_{ij} = f(s_{i-1}, h_j)$, e_{ij} represents an alignment model that decoder at step i which parts of hidden state of the input sequence to be attended. The alignment model f can be a simple feed-forward neural network. Then, the context vector c at step i should be updated on the sequence length T_x as Equation 5.2.

$$c_{ij} = \sum_{j=1}^{T_x} \alpha_{ij} h_j \tag{5.2}$$

The entire process of the seq2seq model using for automatic scripts generation is shown in Figure 5.4.

5.4 Experiments

5.4.1 Datasets

Figure 5.5: Collector and labeling framework: *Wear* (smart-watch) *Hand* (smart-phone).

We evaluate our proposed method using datasets of activity daily livings from Sztyler et al. (2016), which are collected from seven subjects, that recorded manually their daily routine including posture, activity, and location for several days. Figure 5.6 is depicted for the sample rows of the original dataset. Each line of the raw data corresponds to one measurement, in which the first element is the row index; the second element is a

unique time-stamp; the third, fourth and fifth columns respectively denote for 3-axis x, y, and z of accelerometer; the sixth, seventh, eighth, and ninth columns respectively represent for the location, gesture, device position, and activity. The data was collected using

```
id,attr_time,attr_x,attr_y,attr_z,label_environment,label_posture,label_deviceposition,label_activity,label_valid
855990,09.03.15 12:51:21.680,1.0887632,3.3824084,8.870518,Home,Sitting,Thigh,Sport,true
855991,09.03.15 12:51:21.704,1.0809821,3.4356794,8.772355,Home,Sitting,Thigh,Sport,true
855992,09.03.15 12:51:21.720,0.98341835,3.3913867,8.744224,Home,Sitting,Thigh,Sport,true
855993,09.03.15 12:51:21.740,0.95947635,3.3620577,8.725668,Home,Sitting,Thigh,Sport,true
855994,09.03.15 12:51:21.760,0.9648633,3.3446999,8.739435,Home,Sitting,Thigh,Sport,true
855995,09.03.15 12:51:22.816,0.9223662,3.376423,8.826224,Home,Sitting,Thigh,Sport,true
855996,09.03.15 12:51:22.817,0.91458505,3.3746274,8.889671,Home,Sitting,Thigh,Sport,true
855997,09.03.15 12:51:22.818,1.0301052,3.29502,8.738237,Home,Sitting,Thigh,Sport,true
855998,09.03.15 12:51:22.830,0.8972271,3.1639376,9.077017,Home,Sitting,Thigh,Sport,true
855999,09.03.15 12:51:22.841,1.1324574,3.280655,8.904634,Home,Sitting,Thigh,Sport,true
856000,09.03.15 12:51:22.861,1.1228806,3.256713,8.750209,Home,Sitting,Thigh,Sport,true
856001,09.03.15 12:51:22.881,1.1282675,3.1956608,8.780136,Home,Sitting,Thigh,Sport,true
856002,09.03.15 12:51:22.915,1.0426749,3.181894,8.686762,Street/Road/Pasture,Sitting,Thigh,Sport,true
856003,09.03.15 12:51:22.927,0.98461545,3.1627405,8.757391,Street/Road/Pasture,Sitting,Thigh,Sport,true
856004,09.03.15 12:51:22.941,0.96306765,3.1669302,8.874108,Street/Road/Pasture,Sitting,Thigh,Sport,true
856005,09.03.15 12:51:22.961,1.0696096,3.2040405,8.853159,Street/Road/Pasture,Sitting,Thigh,Sport,true
856006,09.03.15 12:51:22.983,1.1085154,3.2501287,9.073426,Street/Road/Pasture,Sitting,Thigh,Sport,true
856007,09.03.15 12:51:23.002,1.0875661,3.1944637,8.944737,Street/Road/Pasture,Sitting,Thigh,Sport,true
856008,09.03.15 12:51:23.021,1.0109516,3.1741128,8.845977,Street/Road/Pasture,Sitting,Thigh,Sport,true
856009,09.03.15 12:51:23.041,0.9864111,3.1711202,8.860941,Street/Road/Pasture,Sitting,Thigh,Sport,true
856010,09.03.15 12:51:23.061,0.9947908,3.1585505,8.923788,Street/Road/Pasture,Sitting,Thigh,Sport,true
856011,09.03.15 12:51:23.083,1.0773908,3.1866825,9.03871,Street/Road/Pasture,Sitting,Thigh,Sport,true
```

Figure 5.6: Sample rows of original dataset.

smart-phones and smartwatches with a self-developed sensor data collector and labeling framework (see Figure 5.5). The framework contains two parts, called *Wear* and *Hand*. The *Wear* application allows updating parameters (posture, activity, and location) immediately, while *Hand* application manages the setting and storing the data (e.g., accelerations, orientation, gps). The data also was recorded simultaneously the acceleration of different body-device position such as the chest, forearm, head, shin, thigh, upper-arm, and waist with a sampling rate of 50Hz. The labels for the mentioned parameters were predefined and could not be changed or extended. Table 5.2 shows the list of three types of semantic concepts that are used for gesture, activity, and location in our experiment.

5.4.2 Experimental Settings and Results

Semantic Concepts Generation Performance

We made all semantic concepts from the signals of wearable sensors. To make concepts, we predicted the labels of sensor data using DNN with the time-sliced of N values. We trained

Gesture	Activity	Location
walking, running, sitting, standing, climbing upstairs, descending downstairs, laying, unknown	desk work, housework, meal preparation, movement, personal grooming, relaxing, shopping, socializing, sport, transportation, unknown	building, home, street, transportation, office, unknown

Table 5.2: List of names of semantic concepts used for gesture, activity and location

our DNN using Adam optimizer Kingma and Ba (2015), with a learning rate of 0.001 and a batch size of 256. We used the cross-entropy loss for a multi-class label of gesture, activity, and location. Then, we predicted labels of semantic concepts with leave-one cross-days training, which trains a model on several days and tested it with another day of data. For the preliminary experiment, we only chose subject 2 from seven subjects in the datasets Sztyler et al. (2016) for our experiment. In subject 2 datasets, the data was collected for thirteen days. In order to test the semantic concepts and scripts generation in one day, we train our DNN on several days (e.g., day 1, 3-9, 12, 13, 15, 16) and test on day 2. The classification performance (F1-score) of gesture, activity, location labels were 0.95, 0.76, 0.94, respectively. As a result, we obtained the labels of 29075 gestures, 29075 activities, 29075 of locations with a time-sliced of 40 and no-overlap between segments from the test set. Afterward, we merged these labels to create semantic concepts. There are two types of concepts that we defined after semantic generation: manually labeled that was collected by the subject and predicted ones in test set, where we consider as TrueConcepts and PredConcepts, respectively. In the next section, we will introduce scripts generation performance from the semantic concepts.

Scripts Generation Performance

We generated a sequence of events in terms of scripts using semantic concepts made from sensor data as presented in the proposed method in section 5.3.2. First, we obtained the manually semantic concepts as described above with 29075 semantic concepts, we consider it as the source language for the encoder of the seq2seq model. Second, we made the target language $(sc_1, sc_2, ..., sc_m)$ for creating a given input decoder of the seq2seq model by using the manual detection templates on extracting semantic concepts. For the training seq2seq model, we used both semantic concepts and target language $(sc_1, sc_2, ..., sc_m)$ of TrueConcepts for encoder and decoder, respectively. For testing seq2seq model, both semantic concepts and target language $(sc_1, sc_2, ..., sc_m)$ of PredConcepts were used.

As a result, we obtained target outputs ($sc_1, sc_2, ..., sc_{m'}$) from test set. Finally, we evaluated target outputs that were generated from the seq2seq model, which is using common metrics: BLEU-1, BLEU-2, BLEU-3 and BLEU-4 in both PredConcepts and TrueConcepts. BLEU is a metric for evaluating a generated sentence to a reference sentence. It is one of the first metrics to claim a high correlation with human judgments of quality. BLEU's output is always a number between 0 and 1. This value indicates how similar the candidate text is to the reference texts, with values closer to 1 representing more similar texts. Cumulative scores refer to the calculation of individual n-gram scores at all orders from 1 to n and weighting them by calculating the weighted geometric mean. The cumulative and individual 1-gram BLEU use the same weights, e.g. (1, 0, 0, 0). The 2-gram weights assign a 50% to each of 1-gram and 2-gram and the 3-gram weights are 33% for each of the 1, 2 and 3-gram scores. The weights for the BLEU-4 are 1/4 (25%) or 0.25 for each of the 1-gram, 2-gram, 3-gram, and 4-gram scores. Figure 5.7 shows our evaluation of PredConcepts and TrueConcepts. As we can see from the Figure, the performances of the seq2seq model on predict labels are high with the BLEU approximate 0.96, while the performances of the seq2seq model on true labels are less than predict ones, especially in case BLEU-4 is 0.76.

	With PredConcepts				With TrueConcepts			
Method	BLEU-1	BLEU-2	BLEU-3	BLEU-4	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Seq2seq +Attention	0.972	0.971	0.967	0.964	0.852	0.838	0.810	0.764

Figure 5.7: Performances of scripts generation using seq2seq model for TrueConcepts and PredConcepts.

Table 5.3 shows an excerpt from the generation of scripts using the seq2seq model. As we can see from the figure, although the seq2seq model can achieve to generate scripts in terms of human-readable text. However, there are repetitive sentences or events in the scripts, hence it is still needed to consider the technique to merge the events (sentences) together.

5.5 Conclusion

In this work, we proposed a novel technique that can generate scripts from human activity recognition using wearable sensor data. As we have known, the HAR systems only allow

The person stands and chats at street or road or pasture.
The person stands and chats at street or road or pasture.
The person stands and chats at street or road or pasture.
The person is at the building.
The person sits and eats or drinks at home.
The person stands and cleans up at home.
The person sits and works at office.
The person sits and works at office.
The person is in the building.
The person is in the building.
The person is in the building.
The person is in the building.
The person is in the building.
The person stands and works at desk in the building.
The person stands and works at desk in the building.
The person stands and works at desk in the building.
The person stands and works at desk in the building.
The person stands and works at desk in the building.

Table 5.3: An excerpt of scripts generation from the model

recognizing discrete activity labels of daily living. Hence, our proposed work shows a step further in HAR application in our daily life. Moreover, the automatically generated scripts from the HAR system enable us to understand what was happened inside numerical data by translating it to human-readable text. Despite the limitation in repetitive events of scripts, hence we need to find out a new technique to merge these events in the future works.

Chapter 6

Learning Scripts from Natural Language Texts

There are difficulties because of a lack of data to extract information from sensor data and it is also very hard to learn events from sensor data. Therefore, in this chapter, we present an approach to learn events from natural language text. In section 6.1 we present an introduction about the scripts learning in Natural Language Processing domain, and its related works in section 6.2. In section 6.3 we show our proposed methods, and the experimental results are illustrated in section 6.4. The chapter ends with a short discussion and an outlook of future work in Section 6.5.

6.1 Introduction

Scripts are thought to be represented in our minds, which present much of our common sense knowledge about the world (such as what usually happens when going to a restaurant). *Script theory* was first introduced by Tomkins (1978) who claims that human behavior mostly follows patterns called *Scripts* which are stereotypical likely sequences of events. Automatically acquiring script models would be useful not only to get a better insight about the human mind but also to build systems which could communicate with human more naturally.

Take for instance the situation where someone says "I am going to a restaurant". The human recipient of this message will naturally infer that this will take some time and that this person will have to choose dishes. This knowledge about the fact that, at a restaurant, people read the menu and wait for their order, is naturally shared between the protagonists of the conversation. However, this implicit knowledge must be made explicit to machines to interpret the situation. Such a script could be learned from the text. Let's take the example of Figure 6.1 where two texts (scenarios 1 and 2 on the right side) describe a

scene of "going at a restaurant". From the two textual scenarios, we would like the overall chain of events (on the left side) to be induced from these texts. If this chain of events can be modeled, such a model can be used to predict events. In fact, prediction of what is likely to happen next given the previous events is a typical use of event chain knowledge. For instance, if the chain of events is known until "X order food" it can be inferred that the next event is likely to be "X eat food".

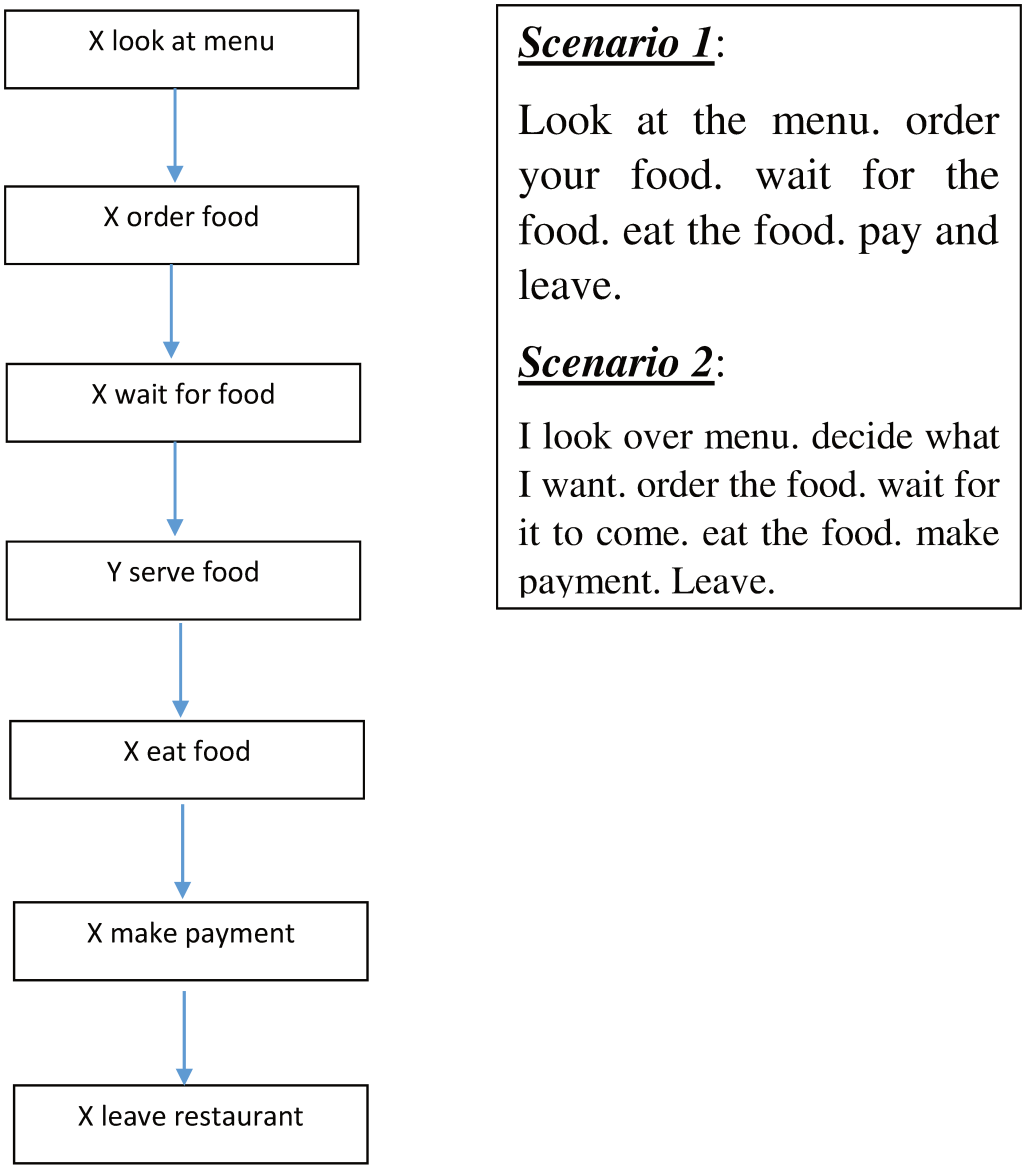


Figure 6.1: Events sequence of restaurant visiting

Hence, such 'script systems' could not only understand an overall situation but could also predict next events or infer likely missing events from a sequence of events. Like other work in the domain, we argue that prototypical scripts can be induced from some

natural language texts [Granroth-Wilding and Clark (2016); Modi (2016); Modi et al. (2017)]. Moreover, this kind of knowledge can be applied to many domains of artificial intelligence, more specifically on automatic narrative generations [A. Baez Miranda et al. (2014); Martin et al. (2018)], information extraction [Dasigi and Hovy (2014); Pichotta and Mooney (2016)], and dialogue Harrison et al. (2016).

Early work on script learning focused on manual construction of script knowledge bases Schank and Abelson (1975), but in recent years there was a strong research interest in the automatic induction of script knowledge from text, for example [Chambers and Jurafsky (2008); Pichotta and Mooney (2014)] investigated studies on naturally occurring texts or crowd-sourced data [Regneri et al. (2010); Li and Riedl (2015)]. Most of these approaches represent the sequence of events as a graph. Each node of the graph is an event, which is modeled as a verbal predicate with its arguments, and arcs correspond to the sequential relation. A graph models common-sense knowledge about the stereotypical sequence of events more than their temporal order[1]. However, most of the recent approaches extracted events from texts using count-based techniques which are usually limited in their modeling due to their poor handling of data sparsity. Indeed, such approaches can capture dependencies only between events that have shared entities (e.g. protagonist) based on probability models acquired from their co-occurrences (count-based method). To avoid this problem, different researchers [Modi (2016); Dasigi and Hovy (2014)] have proposed an event predicate-argument representation based on embedding modeling. The advantage of the event embeddings is to capture the semantic properties of events, i.e., the events with different textual surface forms but which are used in the same context will receive close embeddings. However, one disadvantage of this approach is that it requires prepossessing of the semantics content of the text which is still a challenging task. Furthermore, few approaches exploit the syntactic structure of the event text. For instance, the distance between "John eats the food" and "John is on the table" should be large. Hence when learning the embeddings of "John eats the food on the table" far less weight must be given to the prepositional phrase "on the table".

In this chapter, we present a new technique to model the embeddings of events that is resistant to syntactic variations. This learning is based on Recursive Neural Network (RNN) [2]. This approach is called NESS for recursive neural Network Event Semantic Syntactic, to represent events as a hierarchical composition of their semantic and syntactic predicates

[1]Stereotypical order denotes the order in which events are told in texts not their real temporal order. E.g "I've been to London yesterday. I took the bus" is a pair of events not told in chronological order though perfectly natural.

[2]Recursive Neural Network (RNN) is not to be confused with Recurrent Neural Network (also RNN for short.

and arguments. An advantage of our proposed model is that it can effectively deal with sparsity in semantic space by representing meaning at a higher level of abstraction than the surface forms of words. Moreover, our approach does not rely on extra domain knowledge such as wordnet but only on predefined word embeddings. Results on a publicly available dataset provided by Regneri et al. (2010) shows that one version of *NESS* (*only verbs*) can provide performance superior to the state-of-the-art baseline systems.

Our approach is focused on learning the stereotypical order of events from a set of texts. In that sense, it is a different task than reconstructing temporal ordering (cf. SEM EVAL Bethard et al. (2017)). Furthermore, this modeling does not capture all the information of the script such as the one that can be useful for predicting the next events (see the narrative Cloze Task Chambers and Jurafsky (2008)). However, our model can be extended to model other aspects such as sequence generation, event prediction, and story generation. But we leave these extensions for further work.

Contributions. Our contributions are summarized as follows.

- A technique based on Recursive Neural Network (RNN) for *Scripts* learning, namely *NESS*;
- An experiment with a comparison with baseline models [Regneri et al. (2010); Modi and Titov (2014)] that shows the impact of event embeddings based on syntactic and semantic features.

The chapter is organized as follows. Section 6.2 presents a summary of the state of the art in learning techniques with *Scripts*. *NESS* modeling techniques are detailed in Section 6.3. Experiments are reported in Section 6.4. The chapter ends with a short conclusion and an outlook on future work.

6.2 Related work

Scripts have been popularized in AI by Schank and Abelson (1975) who introduced it as a method for representing procedural knowledge. At that time, event sequences representing scripts were manually encoded in knowledge bases to perform tasks such as event inference, event generation, etc. Nowadays there are a lot of researches on script induction [Chambers and Jurafsky (2008); Regneri et al. (2010); Modi and Titov (2014); Pichotta and Mooney (2014); Jans et al. (2012); Granroth-Wilding and Clark (2016); Pichotta and Mooney (2016); Modi (2016); Hu et al. (2017); Martin et al. (2018)] from texts. Each research developed different approaches which are generally based on two phases: event representation and modeling to learning the scripts.

For **the event representation phase**, the system from Chambers and Jurafsky (2008) presents narrative events as pairs of the form $(event, dependency)$, where the event is represented by a verb and the dependency represents typed dependency relations between event and a protagonist such as subject and object. The event chain is formed by collecting events sharing a common protagonist from texts using a syntactic parser and a co-reference system. The system achieves impressive performance to classify temporal relations between given event descriptions. Another technique introduced by Regneri et al. (2010) considers the whole predicate-argument structure as an atomic unit. But these representations suffer from the sparsity issue which would limit its applicability to machine learning. Another approach from the system of Balasubramanian et al. (2013) proposed Rel-grams to represent events as a triples $(arg_1, relation, arg_2)$, where arg_1 and arg_2 represent the subject and object respectively, to overcome the lack of coherence of protagonist representation of event chains from the system of Chambers and Jurafsky (2008). In Pichotta and Mooney (2014), an event is represented as a tuples of $v(e_s, e_o, e_p)$, where v is a verb lemma, e_s is the subject, e_o is the object, and e_p is an entity with prepositional relation to v. An advantage of multi-arguments event model is that it allows encoding a richer representation of events in comparison with previous methods. Therefore, it has been used in recent works such as [Modi (2016); Granroth-Wilding and Clark (2016)]. Another approach of event representation is that one can use a compositional model based on recursive neural networks Socher et al. (2012), which learns compositional vector representations for phrases and sentences of arbitrary syntactic type and length and which has been shown successfully in detecting atypical events in news Dasigi and Hovy (2014).

For the **sequence modeling phase**, existing models can be considered into two main methods: *weak-order*, and *strong-order*. The former studied in the relations between pairs of events, and the latter investigated the temporal order of events in a full sequence. Event-pair models used discrete event representations and estimated event relations by statistical counting as Chambers and Jurafsky (2008) used Pairwise Mutual Information (PMI) to calculate event relations, and while most sub-sequence models followed Jans et al. (2012) used skip n-gram. However, counting-based methods suffer from data or event sparsity, therefore more recent work developed *embeddings* system to tackle this issue. According to the system of Granroth-Wilding and Clark (2016) leveraged the skip-gram model of Mikolov et al. (2013) for training embedding of events and arguments by ordering them into a pseudo sentence. Modi (2016) applied *word embedding* to verbs and arguments directly and automatically unified event embedding into a single structured event embeddings by using a neural network. Another approach is presented by Frermann et al. (2014) which is based on a hierarchical Bayesian model to build a generative model for *joint* learning of event types and ordering constraints.

In our approach, we adapt the event representation of Socher et al. (2012) to include multiple arguments in an event using a recursive neural network. We then build on Dasigi and Hovy (2014) to learn the stereotypical order of events from the text. Contrary to Dasigi and Hovy (2014) who were working on the detection of atypical news events, we apply it to script learning and we use an explicit neural sequence model (here an LSTM). Another distinction with Dasigi and Hovy (2014) is that our event representation does not need semantic role labeling but only syntactic parsing which is less prone to error. Finally, the semantic distance between events is captured through the recursive word embeddings. Hence the name of our system Neural Event Semantic Syntactic (NESS).

6.3 Proposed Method

In order to capture the order of events from a given sequence of events. The problem can be defined as follows: Given a list of events $E = \{e_1, ..., e_n\}$, provided in the stereotypical order (e.g. $e_i \rightarrow e_j$ if $i < j$), the task is to classify whether two events follow each other with respect to the underlining script. For instance, in the example "waiting for the food" is followed by "eating the food" but not by "leaving the restaurant". Hence, the task is to learn how to classify each pair of events has having a `FOLLOW/NOT_FOLLOW` relation.

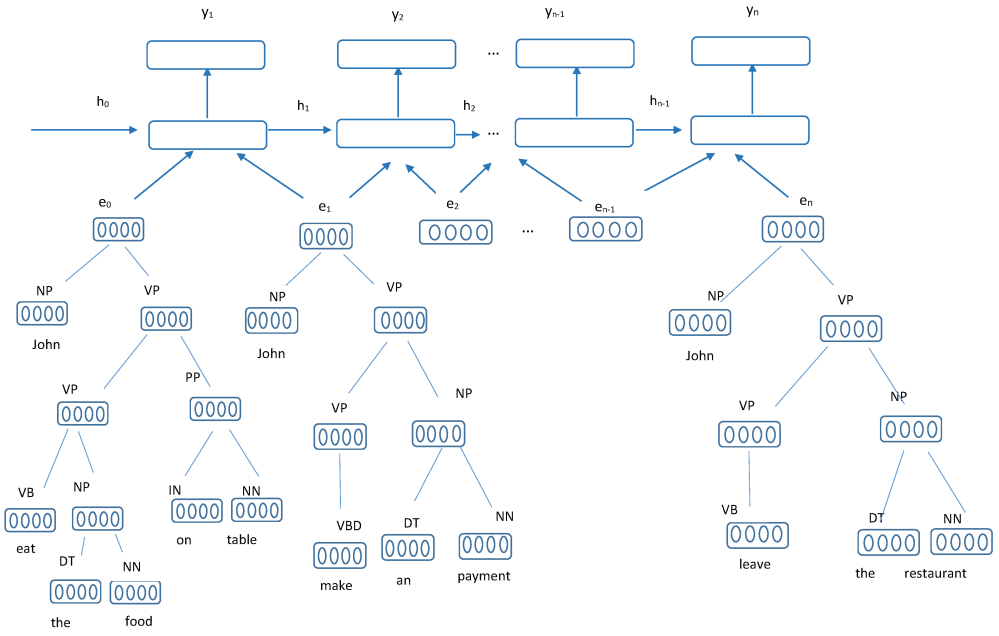


Figure 6.2: NESS model

6.3.1 Learning Model

The model used for learning is based on the Recurrent Neural Network architecture which encodes the input sequence into the fixed-length vector. This model is able to treat a sequence of variable size and has become the standard approach for many tasks in particular in Natural Language Processing tasks Sutskever et al. (2014a). Briefly, a recurrent unit, at each step t takes an input x_t and a previously hidden state h_{t-1} and compute its hidden state and the output using:

$$h_t = \sigma_h(W_h x_t + U_h h_{t-1} + b_h),$$

$$y_t = \sigma_y(W_y h_t + b_y),$$

where y_t is the output vector at each step; W, U, b are the parameters of the neural layer and σ_h and σ_y the activation functions of the neural layers. Numerous improvements have been made to this architecture such as using mono or multi layer of Long Short-Term Memory (LSTM) Hochreiter and Schmidhuber (1997) to prevent the exploding/vanishing gradient problem and to model long dependencies in the sequence.

An LSTM is used to map a sequences of existing events $\{e_1, ..., e_n\}$ into hidden vectors $\{h_1, ..., h_n\}$, which encode the order. Pair of hidden vector linked to 2 events are merged to form u is computed as a composition of arguments (cf. section 6.3.2). This is then passed to a feed forward layer which in turn feed the output layer y which is composed of a single neuron and softmax activation function to make a binary classification (follow or not follow).

6.3.2 Event Representation

Input events are a sequence of natural language sentences. In order to represent an event in a numerical space, a sentence is parsed as a syntactic tree where each node is labeled with its linguistic constituent and the leave by their POS tag (part-Of-Speech). For instance, as exemplified in Figure 6.2, the first event e_0 is "John eats the food on the table". All the word will be attributed their POS tag: "John (NN) eats (VB) the (DT) food (NN) on (IN) the (DT) table (NN)" where NN is a noun, DT is determiner, VB is a verb and IN is a preposition. Higher-level nodes are constituents, for instance `(PP on table)` means that "on table" is a prepositional phrase. Hence the complete syntactic structure of the sentence is fleshed out.

To transform this syntactic structure into a numerical vector representation, we take an argument representation approach. Using this approach, the embeddings of the event can be represented as a single vector e, which is composed of representations of individual words that are guided by the POS tagging.

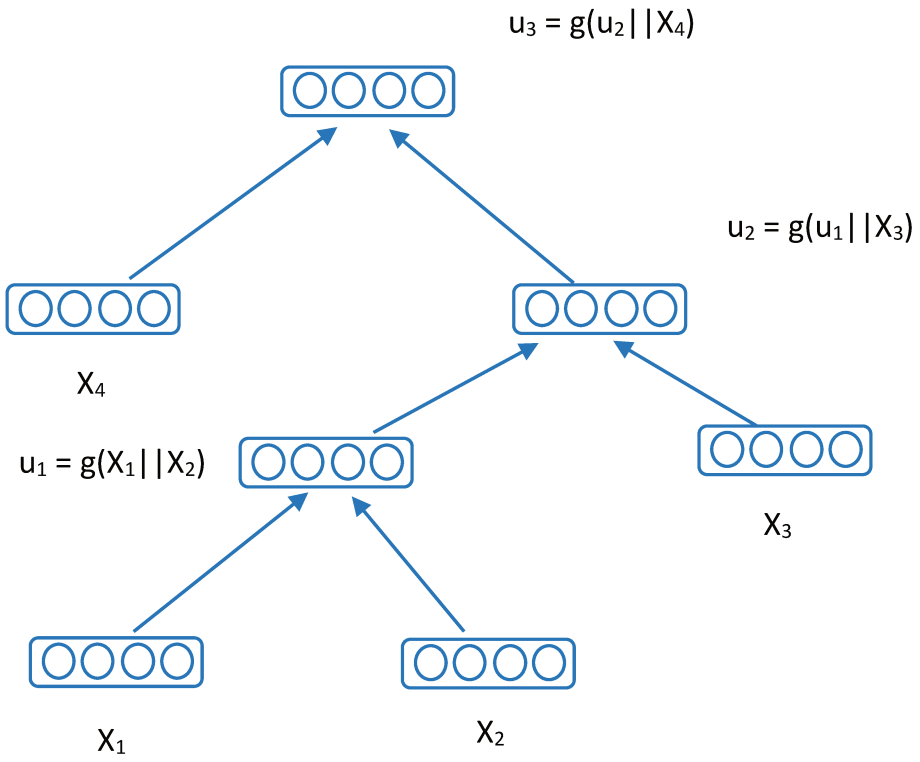


Figure 6.3: Composition of argument backed by a binary tree

An argument composition node takes inputs of dimensionality $2n$ and produces a composed output representation of dimensionality n and a composition score. Figure 6.3 shows an example structure of a binary tree where each non-leaf node has attributed a vector u which is the result of the composition of its two children following equation 6.1.

$$u = g(X_1||X_2) = tanh(W(X_1||X_2) + b) \tag{6.1}$$

In the NN implementation, the g function is a $tanh$ function and $||$ is the vector concatenation operator. During the learning the objective is to learn the parameters $\theta_{arg} = \{W_{arg} \in \mathbb{R}^{n \times 2n}, b_{arg}, S_{arg} \in \mathbb{R}^{n \times 1}, V\}$ where $W_{arg}, b_{arg}, S_{arg}$ are the composition weight, bias and scoring operators respectively, and V is the set of representations of all the words in the vocabulary. We use the vector representation of the words in our vocabulary using the global vectors Pennington et al. (2014).

In this recursive architecture, all nodes use the same parameters θ_{arg}.

The composition score s is a score that shows how good the arguments composition is computed by Equation 6.2. In fact, arguments composition is an unsupervised training approach that Collobert et al. (2011) used to train the RNN for semantic composition based on the contrastive estimation technique proposed by Smith and Eisner (2005), and assuming that any word and its context is a positive example and a random word in the

same context is a negative training example.

$$s = S^T u \tag{6.2}$$

The optimization of θ_{arg} is obtained by Equation 6.3:

$$arg \min_{\theta_{arg}} max(0, 1 - s_{e_i} + s_{e_j} + s_u) \tag{6.3}$$

where s_{e_i} and s_{e_j} are the scores of the composition of the entire argument produce by the root node of argument as the respectively to events e_i and e_j, and s_u is the score produced by randomly replacing one of the words in the argument at a time.

Socher et al. (2013) used a supervised objective that is based on the label error at the topmost node in the RNN. Event composition takes argument representation and produces the event representation and label indicating whether the event is in order or not. Therefore, the event composition node's parameters $\theta_{event} = \{W_{event} \in \mathbb{R}^{n \times kn}, b_{event}, LB_{event} \in \mathbb{R}^{n \times 1}\}$ where k is the number of arguments per event. LB_{event} is the label operator. The objective of this phase is as Equation 6.4:

$$arg \min_{\theta_{event}} (-\alpha \log h(e_i) \times h(e_j) + ((1 - \alpha) \log(1 - h(e_i) \times h(e_j))) \tag{6.4}$$

where α is the reference binary label determining whether the event e_i follows the event e_j or not, and the $h(e_i)$ is the output of hyperbolic tangent function define as following:

$h(e_i) = \frac{1}{1+e^{-LB^T_{event} e_i}}$

We implement the functions and using mini-batch (size = 128) stochastic gradient descent with adam learning Kingma and Ba (2015) schedule.

6.4 Experimental results

6.4.1 Data

To evaluate the approach, we selected the crowd-sourced data provided by Regneri et al. (2010). They collected events sequence descriptions (ESDs) of different kinds of human daily activities (e.g. visit a restaurant, cooking eggs, etc.). Although this dataset is small (only 30 ESDs per activity) it is the only one available to test script learning. The events were extracted from the original dataset using an automatic method for marking up temporal relations in natural language texts Verhagen et al. (2005). These temporal relations follow TimeML (www.timeml.org). The TimeML scheme flags tensed verbs, adjectives, and nominals with EVENT tags with various attributes such as the class of event, tense, grammatical aspect, part-of-speech, and cardinality of the event if it has appeared more than one. For temporal relation annotation, there are 14 temporal relations in TLINK

RelTypes, which reduce to a disjunctive classification of 6 temporal relations where *RelTypes* = {SIMULTANEOUS, AFTER, BEFORE, BEGINS, ENDS, INCLUDES}. An event is SIMULTANEOUS with another event if they appear at the same time interval. In order to annotate the event relations, we only use the BEFORE, AFTER and SIMULTANEOUS to label our events. An excerpt from the dataset resulting from the pre-processing using TimeML framework in combination with our own pre-processing library is shown in Figure 6.4.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
- <scripts>
  - <label type="FOLLOW_UP">
    - <item text="Look" tense="INFINITIVE" pos="VB" aspect="NONE">
        <ptcp text="at" pos="IN"/>
        <ptcp text="the" pos="DT"/>
        <ptcp text="menu" pos="NN"/>
      </item>
    - <item text="Order" tense="INFINITIVE" pos="VB" aspect="NONE">
        <ptcp text="your" pos="PP$"/>
        <ptcp text="food" pos="NN"/>
      </item>
    </label>
  - <label type="FOLLOW_UP">
    - <item text="Order" tense="INFINITIVE" pos="VB" aspect="NONE">
        <ptcp text="your" pos="PP$"/>
        <ptcp text="food" pos="NN"/>
      </item>
    - <item text="wait" tense="INFINITIVE" pos="VB" aspect="NONE">
        <ptcp text="for" pos="IN"/>
        <ptcp text="the" pos="DT"/>
        <ptcp text="food" pos="NN"/>
      </item>
    </label>
```

Figure 6.4: Excerpt of the dataset provided by Regneri et al. (2010) after pre-processing by using TimeML framework and our own preprocessing library.

6.4.2 Baseline methods

As said in the reference section, several methods have been defined to extract the sequence of events. We will use BL, MSA Regneri et al. (2010), BS Frermann et al. (2014), EE_{verb} Modi and Titov (2014), and full model (EE) Modi and Titov (2014) as baseline methods for comparison with NESS. Firstly, verb-frequency baseline (**BL**) chooses the order of events based on the preferred order of the corresponding verbs in the training set: (e_1, e_2) is predicted to be in the stereotypical order if the number of times the corresponding verbs v_1 and v_2 appear in this order in the training event sequence descriptions (ESDs) exceeds the number of times they appear in the opposite order (not necessary at adjacent positions); ties (or if v_1 and v_2 are the same verbs) are broken using random choice.

Another method, Multiple Sequence Alignment (**MSA**) was proposed in Regneri et al. (2010) inspired by research in DNA alignment. The input is some sequences $s_1, ...s_n \in \sum^*$ over some alphabet $\sum$, along with cost function $c_m : \sum \times \sum \leftarrow R$ for substitution and $gapcostc_{gap} \in R$ for insertion and deletion. $\sum$ contains the individual event descriptions while sequence are ESDs. An MSA is a matrix where column are sequences with possibly some gap ϕ between the events of the sequence. Thus each row contains at least a non-gap and if a row contains two events, these are said to be aligned. The alignment is performed using a cost function: $c(A) = c_{gap}.\sum_{\emptyset} + \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=j+1}^{m} c_m(a_{ji}, a_{ki})$, where $c_{gap} \in \mathbb{R}$ is a gap cost for insertion and deletion, $\sum_{\emptyset}$ is the number of gap in A, n is the number of rows and m the number of sequences. If a row contains non-gaps, then these symbols is aligned; aligning a non-gap with a gap can be thought of as an insertion or deletion. In the original implementation, the alignment problem is solved by first aligning two sequences, considering the result as a single sequence whose elements are pairs, and repeating this process until all sequences are incorporated in the MSA. We used the same implementation.

The hierarchical Bayesian model (**BS**) was introduced by Frermann et al. (2014). It is based on the Generalized Mallows Model (GMM) a statistical model over orderings. It takes two parameters σ the canonical ordering and $\rho > 0$ a dispersion parameter which is a penalty for the divergence $d(\pi, \sigma)$ between an observed ordering π and the canonical ordering σ. The Generalized Mallows Model is defined as $GMM(\pi, \sigma) \approx \Pi_i e^{-\rho_i v_i}$ where ρ_i is the *item specific* dispersion parameter and v_i is a vector of inversion counts. The authors then assumed that for each ESD the event e is realized or not by drawing from $Binomial(\Theta_e)^2$. Then, drawing an event ordering π based on the $GMM(\rho)$.

Finally, in Modi and Titov (2014) Event Embeddings (**EE**) are computed based on their predicates and arguments. Given an event, $e = (v, d, a_1, a_2)$, (here v is the predicate lemma, d the dependency and a_1, a_2 are corresponding argument lemmas), each lemma (and dependency) is mapped to a vector using a lookup matrix C. Then each event representation is learned by $A * tanh(T * C_{a_1,:} + R * C_{v,:} + T * C_{a_2,:}) + b$. then the event representations are used in a linear ranker to predict the expected ordering of events. Both the parameters of the compositional process for computing the event representation ((R, T, A and lookup matrix C) and the ranking component of the model are estimated from data.

6.4.3 Results

NESS was also compared to a version of our model which uses only verbs ($NESS^2$) in a similar way to BL and EE_{verbs} in Modi and Titov (2014). The training sets are the *dev set*

Table 6.1: Results on the datasets Regneri et al. (2010) for the verb-frequency baseline (BL), the verb-only embedding model (EE_{verb}), Regneri et al. (2010) (MSA), Frermann et al. (2014) (BS), Modi and Titov (2014) full model (EE), the full model $NESS^1$ and verb-only embedding model ($NESS^2$). Results with models other than NESS are extracted from Modi and Titov (2014)

Test sets	Size	Precision							Recall							F1						
		BL	EE_v	MSA	BS	EE	NESS[1]	NESS[2]	BL	EE_v	MSA	BS	EE	NESS[1]	NESS[2]	BL	EE_v	MSA	BS	EE	NESS[1]	NESS[2]
Bus	276	70.1	81.9	80.0	76.0	85.1	75.2	76.7	71.3	75.8	80.0	76.0	91.9	96.2	98.2	70.7	78.8	80.0	76.0	88.4	84.4	86.1
Coffee	340	70.1	73.7	70.0	68.0	69.5	63.2	64.1	72.6	75.1	78.0	57.0	71.0	96.3	94.6	71.3	74.4	74.0	62.0	70.2	76.2	76.3
Fastfood	236	69.9	81.0	53.0	97.0	90.0	78.9	78.4	65.1	79.1	81.0	65.0	87.9	94.5	95.6	67.4	80.0	64.0	78.0	88.9	86.0	86.1
Return	156	74.0	94.1	48.0	87.0	92.4	77.7	75.9	68.6	91.4	75.0	72.0	89.7	82.3	86.3	71.0	92.8	58.0	79.0	91.0	80.0	80.1
Iron	276	73.4	80.1	78.0	87.0	86.9	68.4	69.3	67.3	69.8	72.0	69.0	80.2	86.0	96.8	70.2	69.8	75.0	77.0	83.4	76.2	80.1
Microw.	450	72.6	79.2	47.0	91.0	82.9	64.8	65.7	63.4	62.8	83.0	74.0	90.3	97.1	99.4	67.7	70.0	60.0	82.0	86.4	77.6	78.8
Eggs	370	72.7	71.4	67.0	77.0	80.7	83.4	80.4	68.0	67.7	64.0	59.0	76.9	91.2	96.7	70.3	69.5	66.0	67.0	78.7	87.2	87.8
Shower	346	62.2	76.2	48.0	85.0	80.0	74.5	76.8	62.5	80.0	82.0	84.0	84.3	99.2	98.4	62.3	78.1	61.0	85.0	82.1	85.2	86.2
Phone	272	67.6	87.8	83.0	92.0	87.5	79.2	82.2	62.8	87.9	86.0	87.0	89.0	99.0	99.1	65.1	87.8	84.0	89.0	88.2	87.9	89.9
Vending	260	66.4	87.3	84.0	90.0	84.2	68.0	67.2	60.6	87.6	85.0	74.0	81.9	95.2	99.1	63.3	84.9	84.0	81.0	88.2	79.3	79.9
Average		69.9	81.3	65.8	85.0	83.9	73.3	73.7	66.2	77.2	78.6	71.7	84.3	94.3	96.4	68.0	79.1	70.6	77.6	84.1	82.0	83.2

(doorbell, laundry, omelet, restaurant), the test sets are 10 sets (bus, coffee, fast-food, return, iron, microwave, eggs, shower, phone, vending). The results are presented in Table 6.1. From these results we can see that NESS is competitive in classifying the order of events. NESS[2] with a verb only reaches an F1-score of 83.2% which outperforms the BL and EE_{verbs} models with an F1-score of 68.0% and 79.1% respectively. In addition, regarding the full model including predicate and arguments, $NESS^1$ achieved 82.0% F1-score higher than the MSA and BS model that reached 70.6% and 77.6% of F1-score. Yet $NESS^1$ is slightly lower than EE which obtained 84.1% of F1-score.

6.5 Conclusion

In this work, we introduced a novel technique based on Recursive Neural Network to modeling the *Scripts* learning in order to capture the order of sequence of events from natural text. We also presented experimental results of our model on the public dataset provided by Regneri et al. (2010), and compare our model with baseline models such as MSA, BL, BS, EE_{verbs}, full model EE that presented the results in Modi and Titov (2014). From the results, we can see that our model can achieve with high accuracy (82.0% and 83.2% in F1-score for both model $NESS^1$ and $NESS^2$ respectively) to learning the order of sequence of events by using RNN model. Moreover, in order to overcome the sparsity issue by using count-based methods we presented the binary tree event embeddings. In future work, we can develop and apply other techniques to learn *Scripts* in order to improve the

semantic representation by using some kind of Tree-Structured Long Short Term Memory Networks Tai et al. (2015).

Chapter 7

Conclusion and Perspectives

7.1 Conclusion

In this thesis, story generation from ambient sensors is approached by a pipeline of abstraction: from raw data to a sequence of sentences summarizing the 'story' of this data. The first step of the pipeline focused on Human Activity Recognition from smartphone data. Indeed, daily life story being about what people do, being able to extract human activity is a necessary step to reach this goal. We approached HAR by machine learning. However, conventional machine learning approaches are sensitive to the problem of imbalanced data. This why our first contribution was a framework for solving the imbalanced data issue. This method enhanced the performance of the conventional machine learning model (e.g., Multilayer Perceptron) for activity classification tasks. However, conventional machine learning models rely on feature engineering to extract parameters from raw data. Therefore, we subsequently used a deep learning approach in order to recognize high-level human activities from raw data. Furthermore, the final text was also generated using a sequence-to-sequence deep model using a concept-to-text approach Qader et al. (2018). These two models used together gave an initial system that can automatically generate scripted texts from wearable sensor data using HAR. Secondly, we introduce an event presentation of script modeling from natural language text on the Natural Language Processing (NLP) domain, which enables us to classify the order of sequence of events so as to predict what event happened next in natural language text. Moreover, with this approach, the model can also predict missing event Modi (2016), and generate sub next event in natural text Hu et al. (2017). However, we let this work for future research. Finally, most current state-of-the-art on both domains of the HAR system and script generation (SG) on NLP has not provided any approach to generate story/script from sensors data. Therefore, different from currently state-of-the-art, we propose a novel approach to invent a system, which allows generating script from HAR using wearable sensors data.

The two main reasons why we have to connect from the HAR domain to script generation on the NLP domain: (1) firstly, due to the segmentation of raw sensors data in order to build the input, which then is fed into the input layer of deep learning for HAR. It leads to the issue of an event would be missing within the sequence of events. As a result, prediction of missing events in the sequence of events by using script modeling will assist to solve this problem; (2) secondly, since the limitation of raw signals recording using sensors, the training, and test data are also limited for prediction tasks on HAR system. Therefore, the prediction of an event using script learning from the natural text would replace the limited raw sensor data, and it also provides a new approach for the HAR task.

In summary, this thesis contributes novel approaches to support for limitations of both HAR and NLP domains. For the HAR domain, script learning can assist the HAR task by predicting missing events and the next event in the sequence of events. For script learning, the HAR task provides an approach to generate a script using deep learning from sensors data. Moreover, the thesis also provides an approach to connect from HAR to the NLP domain by generating script from sensors data.

7.2 Further work and open challenges

Although the thesis has achieved preliminary results to automatically generate scripts/story from wearable sensor data using HAR. Nevertheless, there are issues and open challenges we would like to address for future work.

Firstly, as mentioned in the conclusion of chapter 5, our script generation method using deep learning from wearable sensor data has still suffered from repetitive events. This problem happened because of the segmentation method of raw signals from sensor data. Consequently, we have to find a relevant way to deal with this issue.

Secondly, although we can achieve to use seq2seq with an attention model, which allows the generating model to choose the right words to generate. In addition, it also enriches the semantic and vocabulary of the targeted sentence. However, the corpus of source and target languages are very limited in this thesis, especially in the case of HAR from sensor data. Therefore, we need to find a relevant approach to build richer vocabulary for both source and target language.

Thirdly, it is also interesting to engage other information into our overall approaches such as emotion (e.g., sad, happy, etc.) and health status (e.g., fever, hypothermia, etc.). Obviously, this information would combine with human daily activity and their location so as to make audiences deeply understand the story. Moreover, with this combined information our current semantic concepts daily activity location also can be naturally enriched, which would solve for the limitation of semantic concepts in our current issues.

Therefore, emotion recognition [Tarnowski et al. (2017); Chaparro et al. (2018)] from facial expression or [Batbaatar et al. (2019); Shaheen et al. (2014)] from the text are extensions for our future works. Besides, health status detection Chen and Meng (2011) from physiological monitoring also can be considered as another open challenge for our future works. Furthermore, the health status may assist us to deeper understand human decision-making, which relates to the human context. For instance, the person forgot his jacket during the mountain climbing, his body temperature went down quickly. Therefore, he decided to come back instead of going up to the peak.

Finally, we would like to point out the usefulness of our thesis to the real world. First of all, the thesis gives a novel approach to translate sensors data (e.g., numeric and binary) to text in term of script, which allows audiences to an insight deeply about the sequence of events in the daily activity domain (e.g., smart-home, outdoor environment). It can be applied to health-monitoring in the home environment for elderly people, in which the scripts would be summarized human activities. From the analysis of historical script or summarized activities, it can assist to detect routine behaviour for elderly people. From routine behaviour analysis, we can develop a recommended system to encourage human activity changing. Moreover, we also can investigate emergency alarming system in order to warn unusual changing activity.

Bibliography

A. Baez Miranda, B., Caffiau, S., Garbay, C., and Portet, F. (2014). Task based model for récit generation from sensor data: an early experiment. In *5th International Workshop on Computational Models of Narrative*, pages 1–10.

Abidine, M. B. and Fergani, B. (2014). A new multi-class WSVM classification to imbalanced human activity dataset. *JCP*, pages 1560–1565.

Aker, A. and Gaizauskas, R. J. (2010). Generating image descriptions using dependency relational patterns. In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*, pages 1250–1258.

Altun, K. and Barshan, B. (2010). Human activity recognition using inertial/magnetic sensor units. In *Human Behavior Understanding, First International Workshop, HBU 2010, Istanbul, Turkey, August 22, 2010. Proceedings*, pages 38–51.

Andersen, P. M., Hayes, P. J., Huettner, A. K., Schmandt, L. M., Nirenburg, I. B., and Weinstein, S. P. (1992). Automatic extraction of facts from press releases to generate news stories. In *Proceedings of the Third Conference on Applied Natural Language Processing*, ANLC '92, pages 170–177, Stroudsburg, PA, USA. Association for Computational Linguistics.

Anguita, D., Ghio, A., Oneto, L., Parra, X., and Reyes-Ortiz, J. L. (2012). Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In *Proceedings of the 4th International Conference on Ambient Assisted Living and Home Care*, IWAAL'12, pages 216–223, Berlin, Heidelberg. Springer-Verlag.

Anguita, D., Ghio, A., Oneto, L., Parra, X., and Reyes-Ortiz, J. L. (2013). Energy efficient smartphone-based activity recognition using fixed-point arithmetic. *J. UCS*, 19(9):1295–1314.

Arifoglu, D. and Bouchachia, A. (2017). Activity recognition and abnormal behaviour detection with recurrent neural networks. In *14th International Conference on Mobile*

Systems and Pervasive Computing (MobiSPC 2017) / 12th International Conference on Future Networks and Communications (FNC 2017) / Affiliated Workshops, July 24-26, 2017, Leuven, Belgium, pages 86–93.

Balasubramanian, N., Soderland, S., Mausam, and Etzioni, O. (2013). Generating coherent event schemas at scale. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013*, pages 1721–1731.

Bao, L. and Intille, S. S. (2004). Activity recognition from user-annotated acceleration data. In *Pervasive Computing, Second International Conference, PERVASIVE 2004, Vienna, Austria, April 21-23, 2004, Proceedings*, pages 1–17.

Batbaatar, E., Li, M., and Ryu, K. H. (2019). Semantic-emotion neural network for emotion recognition from text. *IEEE Access*, 7:111866–111878.

Bayat, A., Pomplun, M., and Tran, D. A. (2014). A study on human activity recognition using accelerometer data from smartphones. *Procedia Computer Science*, 34(Supplement C):450 – 457. The 9th International Conference on Future Networks and Communications (FNC'14)/The 11th International Conference on Mobile Systems and Pervasive Computing (MobiSPC'14)/Affiliated Workshops.

Bengio, Y. (2013). Deep learning of representations: Looking forward. In *Statistical Language and Speech Processing - First International Conference, SLSP 2013, Tarragona, Spain, July 29-31, 2013. Proceedings*, pages 1–37.

Berchtold, M., Budde, M., Gordon, D., Schmidtke, H. R., and Beigl, M. (2010a). Actiserv: Activity recognition service for mobile phones. In *14th IEEE International Symposium on Wearable Computers (ISWC 2010), 10-13 October 2010, Seoul, Korea*, pages 1–8.

Berchtold, M., Budde, M., Gordon, D., Schmidtke, H. R., and Beigl, M. (2010b). Actiserv: Activity recognition service for mobile phones. In *14th IEEE International Symposium on Wearable Computers (ISWC 2010), 10-13 October 2010, Seoul, Korea*, pages 1–8.

Bethard, S., Savova, G., Palmer, M., and Pustejovsky, J. (2017). Semeval-2017 task 12: Clinical tempeval. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 565–572.

Bevilacqua, A., MacDonald, K., Rangarej, A., Widjaya, V., Caulfield, B., and Kechadi, M. T. (2018). Human activity recognition with convolutional neural networks. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2018, Dublin, Ireland, September 10-14, 2018, Proceedings, Part III*, pages 541–552.

Biswas, A. and Jacobs, D. W. (2012). Active image clustering: Seeking constraints from humans to complement algorithms. In *CVPR*, pages 2152–2159.

Blachon, D., Coskun, D., and Portet, F. (2014). On-line context aware physical activity recognition from the accelerometer and audio sensors of smartphones. In *AmI*, volume 8850 of *Lecture Notes in Computer Science*, pages 205–220. Springer.

Britz, D., Guan, M. Y., and Luong, M. (2017). Efficient attention using a fixed-size memory representation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 392–400.

Chahuara, P., Fleury, A., Portet, F., and Vacher, M. (2016). On-line Human Activity Recognition from Audio and Home Automation Sensors: comparison of sequential and non-sequential models in realistic Smart Homes. *Journal of ambient intelligence and smart environments*, 8(4):399–422.

Chambers, N. and Jurafsky, D. (2008). Unsupervised learning of narrative event chains. In *ACL 2008, Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, June 15-20, 2008, Columbus, Ohio, USA*, pages 789–797.

Chaparro, V., Gomez, A., Salgado, A., Quintero, O. L., López, N., and Villa, L. F. (2018). Emotion recognition from EEG and facial expressions: a multimodal approach. In *40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2018, Honolulu, HI, USA, July 18-21, 2018*, pages 530–533.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: Synthetic minority over-sampling technique. *J. Artif. Int. Res.*, pages 321–357.

Chen, D. and Meng, M. Q. (2011). Health status detection for patients in physiological monitoring. In *33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2011, Boston, MA, USA, August 30 - Sept. 3, 2011*, pages 4921–4924.

Chen, G., Wang, A., Zhao, S., Liu, L., and Chang, C. (2018). Latent feature learning for activity recognition using simple sensors in smart homes. *Multimedia Tools Appl.*, 77(12):15201–15219.

Chen, Y., Yang, J., Liou, S., Lee, G., and Wang, J. (2008a). Online classifier construction algorithm for human activity detection using a tri-axial accelerometer. *Applied Mathematics and Computation*, 205(2):849–860.

Chen, Y.-P., Yang, J.-Y., Liou, S.-N., Lee, G.-Y., and Wang, J.-S. (2008b). Online classifier construction algorithm for human activity detection using a tri-axial accelerometer. *Applied Mathematics and Computation*, 205(2):849 – 860. Special Issue on Advanced Intelligent Computing Theory and Methodology in Applied Mathematics and Computation.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Dasigi, P. and Hovy, E. (2014). Modeling newswire events using neural networks for anomaly detection. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1414–1422.

Dušek, O., Novikova, J., and Rieser, V. (2018). Findings of the E2E NLG Challenge. In *Proceedings of the 11th International Conference on Natural Language Generation*, Tilburg, The Netherlands. arXiv:1810.01170.

Edel, M. and Koppe, E. (2016). Binarized-blstm-rnn based human activity recognition. In *International Conference on Indoor Positioning and Indoor Navigation, IPIN 2016, Alcala de Henares, Spain, October 4-7, 2016*, pages 1–7.

Ertekin, S. (2013). Adaptive oversampling for imbalanced data classification. In *Information Sciences and Systems 2013 - Proceedings of the 28th International Symposium on Computer and Information Sciences, ISCIS 2013, Paris, France, October 28-29, 2013*, pages 261–269.

Ertekin, S., Huang, J., Bottou, L., and Giles, L. (2007a). Learning on the border: Active learning in imbalanced data classification. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, pages 127–136, New York, NY, USA. ACM.

Ertekin, S., Huang, J., and Giles, C. L. (2007b). Active learning for class imbalance problem. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 823–824, New York, NY, USA. ACM.

Farhadi, A., Hejrati, S. M. M., Sadeghi, M. A., Young, P., Rashtchian, C., Hockenmaier, J., and Forsyth, D. A. (2010). Every picture tells a story: Generating sentences from images. In *Computer Vision - ECCV 2010, 11th European Conference on Computer*

Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV, pages 15–29.

Feng, Y. and Lapata, M. (2010). How many words is a picture worth? automatic caption generation for news images. In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*, pages 1239–1249.

Frermann, L., Titov, I., and Pinkal, M. (2014). A hierarchical bayesian model for unsupervised induction of script knowledge. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2014, April 26-30, 2014, Gothenburg, Sweden*, pages 49–57.

Granroth-Wilding, M. and Clark, S. (2016). What happens next? event prediction using a compositional neural network model. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA.*, pages 2727–2733.

Guadarrama, S., Krishnamoorthy, N., Malkarnenkar, G., Venugopalan, S., Mooney, R. J., Darrell, T., and Saenko, K. (2013). Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*, pages 2712–2719.

Ha, S. and Choi, S. (2016a). Convolutional neural networks for human activity recognition using multiple accelerometer and gyroscope sensors. In *2016 International Joint Conference on Neural Networks, IJCNN 2016, Vancouver, BC, Canada, July 24-29, 2016*, pages 381–388.

Ha, S. and Choi, S. (2016b). Convolutional neural networks for human activity recognition using multiple accelerometer and gyroscope sensors. In *2016 International Joint Conference on Neural Networks, IJCNN 2016, Vancouver, BC, Canada, July 24-29, 2016*, pages 381–388.

Hammerla, N. Y., Fisher, J., Andras, P., Rochester, L., Walker, R., and Ploetz, T. (2015). PD disease state assessment in naturalistic environments using deep learning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, pages 1742–1748.

Hammerla, N. Y., Halloran, S., and Plötz, T. (2016). Deep, convolutional, and recurrent models for human activity recognition using wearables. In *Proceedings of the Twenty-*

Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016, pages 1533–1540.

Han, H., Wang, W.-Y., and Mao, B.-H. (2005). Borderline-smote: A new over-sampling method in imbalanced data sets learning. In *Proceedings of the 2005 International Conference on Advances in Intelligent Computing - Volume Part I*, ICIC'05, pages 878–887.

Hanai, Y., Hori, Y., Nishimura, J., and Kuroda, T. (2009). A versatile recognition processor employing haar-like feature and cascaded classifier. In *IEEE International Solid-State Circuits Conference, ISSCC 2009, Digest of Technical Papers, San Francisco, CA, USA, 8-12 February, 2009*, pages 148–149.

Harrison, B., Banerjee, S., and Riedl, M. O. (2016). Learning from stories: using natural communication to train believable agents. In *IJCAI 2016 Workshop on Interactive Machine Learning. New York*.

He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Trans. on Knowl. and Data Eng.*, 21(9):1263–1284.

He, Z. and Jin, L. (2009). Activity recognition from acceleration data based on discrete consine transform and SVM. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, San Antonio, TX, USA, 11-14 October 2009*, pages 5041–5044.

Hinton, G. E., Osindero, S., and Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Hu, L., Li, J., Nie, L., Li, X., and Shao, C. (2017). What happens next? future subevent prediction using contextual hierarchical LSTM. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, pages 3450–3456.

Hunter, J., Freer, Y., Gatt, A., Reiter, E., Sripada, S., and Sykes, C. (2012). Automatic generation of natural language nursing shift summaries in neonatal intensive care: Bt-nurse. *Artificial Intelligence in Medicine*, 56(3):157–172.

Ignatov, A. (2018). Real-time human activity recognition from accelerometer data using convolutional neural networks. *Appl. Soft Comput.*, 62:915–922.

Inoue, S., Ueda, N., Nohara, Y., and Nakashima, N. (2015). Mobile activity recognition for a whole day: recognizing real nursing activities with big dataset. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp 2015, Osaka, Japan, September 7-11, 2015*, pages 1269–1280.

Jacobs, P. S. and Rau, L. F. (1990). Scisor: Extracting information from on-line news. *Commun. ACM*, 33(11):88–97.

Jans, B., Bethard, S., Vulic, I., and Moens, M. (2012). Skip n-grams and ranking functions for predicting script events. In *EACL 2012, 13th Conference of the European Chapter of the Association for Computational Linguistics, Avignon, France, April 23-27, 2012*, pages 336–344.

Jia, Y. (2009). Diatetic and exercise therapy against diabetes mellitus. In *2009 Second International Conference on Intelligent Networks and Intelligent Systems*, pages 693–696.

Jiang, W. and Yin, Z. (2015a). Human activity recognition using wearable sensors by deep convolutional neural networks. In *Proceedings of the 23rd ACM International Conference on Multimedia*, MM '15, pages 1307–1310, New York, NY, USA. ACM.

Jiang, W. and Yin, Z. (2015b). Human activity recognition using wearable sensors by deep convolutional neural networks. In *Proceedings of the 23rd Annual ACM Conference on Multimedia Conference, MM '15, Brisbane, Australia, October 26 - 30, 2015*, pages 1307–1310.

Kashimoto, Y., Morita, T., Fujimoto, M., Arakawa, Y., Suwa, H., and Yasumoto, K. (2017). Sensing activities and locations of senior citizens toward automatic daycare report generation. In *31st IEEE International Conference on Advanced Information Networking and Applications, AINA 2017, Taipei, Taiwan, March 27-29, 2017*, pages 174–181.

Khan, A. M., Lee, Y.-K., Lee, S. Y., and Kim, T.-S. (2010). A triaxial accelerometer-based physical-activity recognition via augmented-signal features and a hierarchical recognizer. *Trans. Info. Tech. Biomed.*, 14(5):1166–1172.

Khan, A. M., Tufail, A., Khattak, A. M., and Laine, T. H. (2014). Activity recognition on smartphones via sensor-fusion and kda-based svms. *IJDSN*, 10.

Kim, Y. and Toomajian, B. (2016). Hand gesture recognition using micro-doppler signatures with convolutional neural network. *IEEE Access*, 4:7125–7130.

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

Kulkarni, G., Premraj, V., Dhar, S., Li, S., Choi, Y., Berg, A. C., and Berg, T. L. (2011). Baby talk: Understanding and generating simple image descriptions. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*, pages 1601–1608.

Kuznetsova, P., Ordonez, V., Berg, A. C., Berg, T. L., and Choi, Y. (2012). Collective generation of natural image descriptions. In *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, July 8-14, 2012, Jeju Island, Korea - Volume 1: Long Papers*, pages 359–368.

Kwapisz, J. R., Weiss, G. M., and Moore, S. A. (2011). Activity recognition using cell phone accelerometers. *SIGKDD Explor. Newsl.*, 12(2):74–82.

Lara, O. D. and Labrador, M. A. (2012). A mobile human activity recognition system. In *2012 IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, NV, USA, January 14-17, 2012*, pages 38–39.

Lara, O. D. and Labrador, M. A. (2013). A Survey on Human Activity Recognition using Wearable Sensors. *IEEE Communications Surveys & Tutorials*, 15(3):1192–1209.

Lara, Óscar D.., Pérez, A. J., Labrador, M. A., and Posada, J. D. (2012). Centinela: A human activity recognition system based on acceleration and vital sign data. *Pervasive and Mobile Computing*, 8(5):717 – 729.

Lee, S. and Mase, K. (2002). Activity and location recognition using wearable sensors. *IEEE Pervasive Computing*, 1(3):24–32.

Li, B. and Riedl, M. O. (2015). Scheherazade: Crowd-powered interactive narrative generation. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 4305–4306.

Li, Y., Shi, D., Ding, B., and Liu, D. (2014). Unsupervised feature learning for human activity recognition using smartphone sensors. In *Mining Intelligence and Knowledge Exploration - Second International Conference, MIKE 2014, Cork, Ireland, December 10-12, 2014. Proceedings*, pages 99–107.

Liu, C., Zhang, L., Liu, Z., Liu, K., Li, X., and Liu, Y. (2016). Lasagna: towards deep hierarchical understanding and searching over mobile sensing data. In *Proceedings of the*

22nd Annual International Conference on Mobile Computing and Networking, MobiCom 2016, New York City, NY, USA, October 3-7, 2016, pages 334–347.

Ma, M., Fan, H., and Kitani, K. M. (2016). Going deeper into first-person activity recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 1894–1903.

Mai, S. T., Assent, I., and Storgaard, M. (2016). AnyDBC: An Efficient Anytime Density-based Clustering Algorithm for Very Large Complex Datasets. In *KDD*, pages 1025–1034.

Martin, L. J., Ammanabrolu, P., Wang, X., Hancock, W., Singh, S., Harrison, B., and Riedl, M. O. (2018). Event representations for automated story generation with deep neural nets. In *AAAI*.

Micucci, D., Mobilio, M., and Napoletano, P. (2017). Unimib shar: a new dataset for human activity recognition using acceleration data from smartphones. In *Applied Sciences*, volume 7.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 3111–3119.

Miranda, B. B. (2018). *Génération de récits à partir de données ambiantes. (Generating stories from ambient data)*. PhD thesis, Grenoble Alpes University, France.

Miyanishi, T., Hirayama, J., Maekawa, T., and Kawanabe, M. (2018). Generating an event timeline about daily activities from a semantic concept stream. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 142–150.

Modi, A. (2016). Event embeddings for semantic script modeling. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016*, pages 75–83.

Modi, A., Anikina, T., Ostermann, S., and Pinkal, M. (2016). Inscript: Narrative texts annotated with script information. In *Proceedings of the Tenth International Conference*

on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016.

Modi, A. and Titov, I. (2014). Inducing neural models of script knowledge. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning, CoNLL 2014, Baltimore, Maryland, USA, June 26-27, 2014*, pages 49–57.

Modi, A., Titov, I., Demberg, V., Sayeed, A., and Pinkal, M. (2017). Modelling semantic expectation: Using script knowledge for referent prediction. *Transactions of the Association for Computational Linguistics*, 5:31–44.

Morales, F. J. O. and Roggen, D. (2016). Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors*, 16(1):115.

Murad, A. and Pyun, J. (2017). Deep recurrent neural networks for human activity recognition. *Sensors*, 17(11):2556.

Nguyen, K. T., Portet, F., and Garbay, C. (2018). Dealing with imbalanced data sets for human activity recognition using mobile phone sensors. In *Intelligent Environments 2018 - Workshop Proceedings of the 14th International Conference on Intelligent Environments, Rome, Italy, 25-28 June 2018*, pages 129–138.

Ohnishi, K., Kanehira, A., Kanezaki, A., and Harada, T. (2016). Recognizing activities of daily living with a wrist-mounted camera. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 3103–3111.

Ordonez, V., Kulkarni, G., and Berg, T. L. (2011). Im2text: Describing images using 1 million captioned photographs. In *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain.*, pages 1143–1151.

Pärkkä, J., Ermes, M., Korpipää, P., Mäntyjärvi, J., Peltola, J., and Korhonen, I. (2006). Activity classification using realistic data from wearable sensors. *IEEE Trans. Information Technology in Biomedicine*, 10(1):119–128.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 1532–1543.

Perez, A. J., Labrador, M. A., and Barbeau, S. J. (2010). G-sense: a scalable architecture for global sensing and monitoring. *IEEE Network*, 24(4):57–64.

Pichotta, K. and Mooney, R. J. (2014). Statistical script learning with multi-argument events. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2014, April 26-30, 2014, Gothenburg, Sweden*, pages 220–229.

Pichotta, K. and Mooney, R. J. (2016). Learning statistical scripts with lstm recurrent neural networks. In *AAAI*, pages 2800–2806.

Pirsiavash, H. and Ramanan, D. (2012). Detecting activities of daily living in first-person camera views. In *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*, pages 2847–2854.

Poibeau, T., Saggion, H., Piskorski, J., and Yangarber, R., editors (2013). *Multi-source, Multilingual Information Extraction and Summarization*. Theory and Applications of Natural Language Processing. Springer.

Portet, F., Reiter, E., Gatt, A., Hunter, J., Sripada, S., Freer, Y., and Sykes, C. (2009). Automatic generation of textual summaries from neonatal intensive care data. *Artificial Intelligence*, 173(7-8):789–816. AvImpFact=2.566 estim. in 2012.

Qader, R., Jneid, K., Portet, F., and Labbé, C. (2018). Generation of company descriptions using concept-to-text and text-to-text deep models: dataset collection and systems evaluation. In *Proceedings of the 11th International Conference on Natural Language Generation, Tilburg University, The Netherlands, November 5-8, 2018*, pages 254–263.

Regneri, M., Koller, A., and Pinkal, M. (2010). Learning script knowledge with web experiments. In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*, pages 979–988.

Reiter, E. and Dale, R. (2000). *Building Natural Language Generation Systems*. Cambridge University Press, New York, NY, USA.

Reiter, E., Sripada, S., Hunter, J., Yu, J., and Davy, I. (2005). Choosing words in computer-generated weather forecasts. *Artif. Intell.*, 167(1-2):137–169.

Riboni, D. and Bettini, C. (2011). Cosar: Hybrid reasoning for context-aware activity recognition. *Personal Ubiquitous Comput.*, 15(3):271–289.

Rohrbach, M., Qiu, W., Titov, I., Thater, S., Pinkal, M., and Schiele, B. (2013). Translating video content to natural language descriptions. In *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*, pages 433–440.

Ronao, C. A. and Cho, S. (2015). Deep convolutional neural networks for human activity recognition with smartphone sensors. In *ICONIP (4)*, volume 9492 of *Lecture Notes in Computer Science*, pages 46–53. Springer.

Sani, S., Wiratunga, N., Massie, S., and Cooper, K. (2017). knn sampling for personalised human activity recognition. In *Case-Based Reasoning Research and Development - 25th International Conference, ICCBR 2017, Trondheim, Norway, June 26-28, 2017, Proceedings*, pages 330–344.

Saputri, T. R. D., Khan, A. M., and Lee, S. (2014). User-independent activity recognition via three-stage ga-based feature selection. *IJDSN*, 10.

Schank, R. C. and Abelson, R. P. (1975). Scripts, plans and knowledge. In *Advance Papers of the Fourth International Joint Conference on Artificial Intelligence, Tbilisi, Georgia, USSR, September 3-8, 1975*, pages 151–157.

Settles, B. (2010). Active learning literature survey. Technical report, University of Wisconsin–Madison.

Shaheen, S., El-Hajj, W., Hajj, H. M., and Elbassuoni, S. (2014). Emotion recognition from text based on automatically generated rules. In *2014 IEEE International Conference on Data Mining Workshops, ICDM Workshops 2014, Shenzhen, China, December 14, 2014*, pages 383–392.

Shannon, C. E. (2001). A mathematical theory of communication. *Mobile Computing and Communications Review*, 5:3–55.

Smith, N. A. and Eisner, J. (2005). Contrastive estimation: Training log-linear models on unlabeled data. In *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA*, pages 354–362.

Socher, R., Huval, B., Manning, C. D., and Ng, A. Y. (2012). Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea*, pages 1201–1211.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014a). Sequence to sequence learning with neural networks. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 3104–3112.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014b). Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Sztyler, T., Carmona, J., Völker, J., and Stuckenschmidt, H. (2016). Self-tracking reloaded: Applying process mining to personalized health care from labeled sensor data. *T. Petri Nets and Other Models of Concurrency*, 11:160–180.

Sztyler, T. and Stuckenschmidt, H. (2016). On-body localization of wearable devices: An investigation of position-aware activity recognition. In *2016 IEEE International Conference on Pervasive Computing and Communications, PerCom 2016, Sydney, Australia, March 14-19, 2016*, pages 1–9.

Tai, K. S., Socher, R., and Manning, C. D. (2015). Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 1556–1566.

Tan, C. C., Jiang, Y., and Ngo, C. (2011). Towards textually describing complex video contents with audio-visual concept classifiers. In *Proceedings of the 19th International Conference on Multimedia 2011, Scottsdale, AZ, USA, November 28 - December 1, 2011*, pages 655–658.

Tapia, E. M., Intille, S. S., Haskell, W. L., Larson, K., Wright, J. A., King, A., and Friedman, R. H. (2007). Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor. In *11th IEEE International Symposium on Wearable Computers (ISWC 2007), October 11-13, 2007, Boston, MA, USA*, pages 37–40.

Tarnowski, P., Kolodziej, M., Majkowski, A., and Rak, R. J. (2017). Emotion recognition using facial expressions. In *International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland*, pages 1175–1184.

Tomkins, S. S. (1978). Script theory: differential magnification of affects. *Nebraska Symposium on Motivation.*, 26:201–236.

Tong, S. and Koller, D. (2001). Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2:45–66.

Vaizman, Y., Ellis, K., and Lanckriet, G. R. G. (2017). Recognizing detailed human context in the wild from smartphones and smartwatches. *IEEE Pervasive Computing*, pages 62–74.

Venugopalan, S., Hendricks, L. A., Mooney, R. J., and Saenko, K. (2016). Improving lstm-based video description with linguistic knowledge mined from text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1961–1966.

Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R. J., Darrell, T., and Saenko, K. (2015a). Sequence to sequence - video to text. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 4534–4542.

Venugopalan, S., Xu, H., Donahue, J., Rohrbach, M., Mooney, R. J., and Saenko, K. (2015b). Translating videos to natural language using deep recurrent neural networks. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 1494–1504.

Vepakomma, P., De, D., Das, S. K., and Bhansali, S. (2015). A-wristocracy: Deep learning on wrist-worn sensing for recognition of user complex activities. In *12th IEEE International Conference on Wearable and Implantable Body Sensor Networks, BSN 2015, Cambridge, MA, USA, June 9-12, 2015*, pages 1–6.

Verhagen, M., Mani, I., Saurí, R., Littman, J., Knippen, R., Jang, S. B., Rumshisky, A., Phillips, J., and Pustejovsky, J. (2005). Automating temporal annotation with TARSQI. In *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA*, pages 81–84.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P. (2008). Extracting and composing robust features with denoising autoencoders. In *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, pages 1096–1103.

Wang, J., Chen, Y., Hao, S., Peng, X., and Hu, L. (2019). Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters*, 119:3–11.

Wang, L. (2016). Recognition of human activities using continuous autoencoders with wearable sensors. *Sensors*, 16(2):189.

Williams, S. and Reiter, E. (2008). Generating basic skills reports for low-skilled readers*. *Nat. Lang. Eng.*, 14(4):495–525.

Yang, J., Nguyen, M. N., San, P. P., Li, X., and Krishnaswamy, S. (2015). Deep convolutional neural networks on multichannel time series for human activity recognition. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 3995–4001.

Yang, Q. (2009). Activity recognition: Linking low-level sensors to high-level intelligence. In *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009*, pages 20–25.

Yao, S., Hu, S., Zhao, Y., Zhang, A., and Abdelzaher, T. F. (2017). Deepsense: A unified deep learning framework for time-series mobile sensing data processing. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pages 351–360.

Yen, S.-J. and Lee, Y.-S. (2009). Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Syst. Appl.*, 36(3):5718–5727.

Zebin, T., Scully, P. J., and Ozanyan, K. B. (2016). Inertial sensor based modelling of human activity classes: Feature extraction and multi-sensor data fusion using machine learning algorithms. In *eHealth 360° - International Summit on eHealth, Budapest, Hungary, June 14-16, 2016, Revised Selected Papers*, pages 306–314.

Zeng, M., Nguyen, L. T., Yu, B., Mengshoel, O. J., Zhu, J., Wu, P., and Zhang, J. (2014a). Convolutional neural networks for human activity recognition using mobile sensors. In *6th International Conference on Mobile Computing, Applications and Services, MobiCASE 2014, Austin, TX, USA, November 6-7, 2014*, pages 197–205.

Zeng, M., Nguyen, L. T., Yu, B., Mengshoel, O. J., Zhu, J., Wu, P., and Zhang, J. (2014b). Convolutional neural networks for human activity recognition using mobile sensors. In *6th International Conference on Mobile Computing, Applications and Services, MobiCASE 2014, Austin, TX, USA, November 6-7, 2014*, pages 197–205.

Zhang, L., Wu, X., and Luo, D. (2015). Real-time activity recognition on smartphones using deep neural networks. In *2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom), Beijing, China, August 10-14, 2015*, pages 1236–1242.

Zheng, Y., Liu, Q., Chen, E., Ge, Y., and Zhao, J. L. (2016). Exploiting multi-channels deep convolutional neural networks for multivariate time series classification. *Frontiers Comput. Sci.*, 10(1):96–112.

Zhu, C. and Sheng, W. (2009). Multi-sensor fusion for human daily activity recognition in robot-assisted living. In *Proceedings of the 4th ACM/IEEE International Conference on Human Robot Interaction*, HRI '09, pages 303–304, New York, NY, USA. ACM.