Volume 38 Number 4 December 2014 ISSN 0350-5596

Informatica

An International Journal of Computing and Informatics

Editorial Boards

Informatica is a journal primarily covering intelligent systems in the European computer science, informatics and cognitive community; scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor from the Editorial Board can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the list of referees. Each paper bears the name of the editor who appointed the referees. Each editor can propose new members for the Editorial Board or referees. Editors and referees inactive for a longer period can be automatically replaced. Changes in the Editorial Board are confirmed by the Executive Editors.

The coordination necessary is made through the Executive Editors who examine the reviews, sort the accepted articles and maintain appropriate international distribution. The Executive Board is appointed by the Society Informatika. Informatica is partially supported by the Slovenian Ministry of Higher Education, Science and Technology.

Each author is guaranteed to receive the reviews of his article. When accepted, publication in Informatica is guaranteed in less than one year after the Executive Editors receive the corrected version of the article.

Editorial: "Michie-Turing" IS2014 Award Recipient: Janez Grad

It is my honour and privilege to introduce the 2014 "Donald Michie and Alan Turing" prize recipient Janez Grad for Life Achievements in Slovenian Information Society (http://is.ijs.si/is_awards.html), aka "Michie-Turing" Award.

The name of the prize was proposed by Stephen Muggleton when we were discussing Slovenian international information-society events. Following a long analysis with colleagues we have set the nomination procedure inspired by the procedure Nobel prizes are awarded. Of course, our award cannot compete with the best world-wide awards like the Turing award, the »Nobel prize« for computing. Nevertheless, we wanted to highlight and emphasise the contributions of Donald Michie and Alan Turing on the one side, and our achievements in the fields of information society, computing, and informatics on the other side. The tuning of the procedure and obtaining the approval from the descendants of Donald Michie and Alan Turing demanded quite some time. At this point I would like to express my gratitude to all colleagues involved. The name "Michie-Turing" Award was coined from Alan Turing, computer »Einstein«, and Donald Michie, Turing's contemporary, who often visited Slovenia and together with Ivan Bratko helped establishing Slovenian AI school. Both names are highly appreciated in our societies - for artificial intelligence SLAIS, for computer science and informatics Informatika, and ACM Slovenia, among others. Donald was a very welcome and frequent guest in Slovenia and I often remember him staying in a room five doors from mine. His room still bears the name »Donald Michie room« and mine »Alan Turing room«. We celebrated their names in a series of events, e.g. at the conference itself, with a special issue of Informatica Vol 37, no. 1, 2013. The editors' Introduction to the Special Issue on "100 Years of Alan Turing and 20 Years of SLAIS" was written by Dunja Mladenić, Stephen Muggleton, and Ivan Bratko. It consisted of the following contributions: C. Sammut: The Child Machine vs. the World Brain; M. Gams: Alan Turing, Turing Machines and Stronger; A. Bifet: Mining Big Data in Real Time; J. Gama: Data Stream Mining: the Bounded Rationality; D. Mladenić and M. Grobelnik: Automatic Text Analysis by Artificial Intelligence; N. Lavrač and P. Kralj Novak: Relational and Semantic Data Mining for Biomedical Research; I. Kononenko, E. Štrumbelj, Z. Bosnić, D. Pevec, M. Kukar, and M. Robnik Šikonja: Explanation and Reliability of Individual Predictions; M. Bohanec, M. Žnidaršič, V. Rajkovič, I. Bratko, and B. Zupan: DEX Methodology: Three Decades of Qualitative Multi-Attribute Modeling; J. Demšar and B. Zupan: Orange: Data Mining Fruitful and Fun - A Historical Perspective.

In 2014, the Slovenian societies supporting Informatica decided to introduce the Michie-Turing prize recipient in a special editorial of this journal. The prize winner Janez Grad provided an overview of his early years in computing, titled:

A Survey of Computer Usage in Education and Research at the University of Ljubljana, with Emphasis on RCC and the UCC

Prof. Grad has been engaged in the process of the computer usage, computer science and informatics since 1960 when he started working at the (Nuclear) Institute »Jožef Stefan« (NIJS, later IJS), Ljubljana. At that time computer usage has first been introduced into processes of some Slovenian business companies, and education and research institutions, respectively, in particular the University of Ljubljana (ULj) and NIJS. In the following, the evolutionary processes of computer usage and the supporting activities in the above stated institutions are described.

In the 1960', the main players in informatics were the INTERTRADE enterprise, Ljubljana, that played the main role in most of the business companies by supplying them with the IBM technology on one side, and on the other side ULj and NIJS, where NIJS was the main initiator and action performer. For the purpose of its research process, NIJS was first using the IBM 705 computer which was installed at the Federal Statistical Office in Belgrade, Yugoslavia (computer programmer at NIJS was Prof. Janez Grad). Later on in 1962, ULj together with NIJS, from here on both together named as ULj, transferred their computer usage to the ZUSE-Z-23 computer system being installed in Ljubljana with the help of the Composite Organization of Associated Labour ISKRA (COAL ISKRA), Kranj, which has started the licence production of this type of computers in cooperation with the German company ZUSE (responsible employees for the system running were Prof. Janez Grad and Mr. Cveto Trampuž). By means of the financial support from ISKRA, the researchers at ULj programmed mathematical algorithms to be used on the Z-23 computer and the licensed computers.

The use of computers at ULj that started in 1961 expanded very rapidly in the following years. In addition to the use in research it has spread for educational purposes to all the faculties and also into the administration departments. All these activities demanded a new, more powerful computer system. But as the total amount of the university work on computers was too small for a justified and rational exploitation of a big computer system, and as ULj did not have the adequate financial means for purchasing a new computer on its own, ULj and in particular IJS gave the initiative for a joint big computer investment shared by ULj, the electronic equipment producers COAL ISKRA, Kranj and Slovenian government institutions, Ljubljana. And so a CDC 3300 computer of an appropriate configuration was installed in 1968. It was running in a batch mode with no terminals. The site of the installed CDC 3300 computer, named as »Republic Computer Centre« (RCC), was on the outskirts of Ljubljana, in Ljubljana – Stegne, a couple of kilometres away from the ULj and other founders' sites - which resulted in poor communications between the RCC and their users. Prof.

Janez Grad was appointed as the head of RCC. Due to these facts, the managers of computer usage at ULj bought another smaller computer, IBM 1130, intended for the use of students, and installed it in the vicinity of the faculties at the Institute of Mathematics, Physics and Mechanics. Research work was processed partly on this computer as well.

Meanwhile, the expansion of computer usage continued at ULj. Besides the usual batch processing, the need for computer usage in process control appeared at the Faculty for Mechanical Engineering and IJS, and hybrid computers were developed by the Faculty for Electrical Engineering. ULj bought four new small computers in order to meet these demands, one IBM 1130, one IBM System 7 and two CDC 1700 computers. In order to cope with all emerging activities and problems accompanied them, ULj formed a group of experts – »The professional committee for computerization«, and named Prof. Dr. Jernej Virant as its head. Shortly after that, in 1971 ULj founded also the »University Computer Centre« (UCC) and named Prof. Janez Grad as the head of UCC. In the years 1971 – 73 they together designed a plan for the university terminal system oriented towards the central computer system in RCC which was accomplished by the middle of 1975. In 1971, RCC decided to buy and operate a more powerful computer system CDC CYBER 70, which enabled multiprogramming and remote job entry. As (acting) directors of RCC were successively named Dr. Edo Pirkmajer, Mr. Cveto Trampuž and finaly Dr. Desan Justin. As the new computer system was too big for the workload of the RCC founders, they invited some additional companies and institutions to join as partners of the extended RCC. And quite few of them really have joined the RCC, for example Educational Community of Slovenia, Research Community of Slovenia, Republic Road Community, National Bank, Ljubljana Dairies, Building Enterprise Obnova, Forestry of Slovenia, etc.

Due to ever increasing number of students of University of Ljubljana, which exceeded over 20.000, who needed a simultaneous computer usage UCC designed a timesharing and interactive type of work in 1981. The network embraced all the faculties and other high schools of ULj, and also the Republic Center for CAI (Computer Aided Instructions) at the Faculty for natural sciences and technology; the head was Prof. Dr. Aleksandra Kornhauser. This required a terminal system with over 100 display and/or teletype terminals and personal computers, respectively. As the CYBER computer at RCC did not quite suit these requirements, UCC decided to install and use a DEC system-10 with DELTA knot computers and KOPA 1000 and LA 34 terminals, respectively. The system has been further extended in the following years, and Mr. Franc Mandelc has been appointed as the head of UCC. ULj has started a new step in computer usage in years 1987-88, when DEC-10 has been exchanged by two VAX 8550 computers and a network of over 200 terminals and personal computers. The JUPAK system of Yugoslav Post Offices enabled the UCC computer system to be connected to other computers in Slovenia, Yugoslavia, and Europe. In this way a direct information exchange between ULj and the universities in Europe became possible through e-mail (BITNET, COSINE) and joint (common) data bases could be built up and used. Nowadays UCC has been serving as ULj information centre supplying faculties and other users the information via university IT network, while the faculties use their own computer equipment. In meantime, RCC has been transformed into a centre used by Slovenian companies for solving their application problems in their business processes. The IT development at Slovenian universities, research institutes and in Slovenian society on the whole, for instance in Slovenian Society Informatika, Ljubljana Economists Society, COAL ISKRA, ISKRA DELTA enterprise, INTERTRADE enterprise, etc., was generally accompanied by intensive endeavour in research, teaching and development activities, which all helped to introduce the computer science and informatics programmes into the study and research programmes of most of the high schools and secondary schools and business sphere as well.

Some activities of the ULj and IJS researchers and teaching staff within the process of the computer system development and its usage at ULj and IJS: Some activities were realized within the frame of some professional associations while others were performed solely within ULj and IJS, respectively. For instance: Organization and realization of professional gatherings (conference, symposium, consultations) such as FCIP, IFIP CONGRESS 71, INFORMATICA, DSI, SOR, IS International Multiconferences (IJS, organizers Prof. Dr. Matjaž Gams, Prof. Dr. Vladislav Rajkovič, …);

Publishing of periodical professional journals such as: (1) INFORMATICA, An international Journal of Computing and Informatics (IJS, Executive Editors Dr. Anton P. Železnikar, Prof. Dr. Matjaž Gams); (2) The Applied INFORMATICS; etc. Publishing research papers, monographs, text books, other books (such as: (1) The Electronic Computers. Editor Franc Spiller-Muys, Iskra, 26 authors, Published on the occasion of the IFIP CONGRESS 71, Ljubljana; (2) Introduction into Computer Science. DZS, Dr. Ivan Bratko, Dr. Vladislav Rajkovič).

Carrying through many research projects (such as: The Foundation of an Information Centre I - III. Dr. Janez Grad and 13 co-authors) and the computer network planning projects (such as: The University Computer Network. UCC, Chief editor Dr. Janez Grad, Executive Editor Mr. Franc Mandelc);

Analysing, describing and publishing the professional profiles in the field of computer usage and informatics (Governmental department, presiding by Prof. Dr. Jernej Virant, 20 co-authors).

(1) Business Informatics Dictionary. Prof. Dr. Ivan Turk and 37 co-workers); (2) Dictionary of Computing. CZ, technical-professional editor Dr. Matjaž Gams, 9 national co-ordinators and 80 co- workers.

Matjaž Gams
Managing Editor, Informatica

Automatic Android-based Wireless Mesh Networks

Paul Wong, Vijay Varikota, Duong Nguyen and Ahmed Abukmail
School of Science and Computer Engineering
University of Houston - Clear Lake, Houston, Tx 77058, USA
Email: wongp7342@uhcl.edu, varikotav6521@uhcl.edu, nguyend4081@uhcl.edu, abukmail@uhcl.edu

Keywords: Android, wireless mesh networks, Wi-Fi Direct, relay

Received: August 14, 2014

Android devices are portable, equipped with powerful processors, and can act as both mesh routers and clients through Wi-Fi Direct, making them a suitable platform from which to deploy a wireless mesh network. To allow Android devices to automatically establish and participate in a wireless mesh network, we circumvent two key problems of the Android's Wi-Fi APIs: lack of direct support for ad-hoc networking / many-to-one Wi-Fi Direct connections, and Wi-Fi Direct WPS authentication limiting applications to ones which require intermittent user interaction.

Povzetek: Opisana je metoda za izboljševanje vključevanja naprav Android v omrežja.

1 Introduction

Devices running Android[1], such as smartphones and tablets, are ubiquitous, power efficient, and inexpensive. In addition, most Android devices come with a suite of sensors which, in a network, may be used to collect physical data simultaneously in different locations; this gives a wireless mesh network containing Android devices the potential to be immediately useful in a variety of applications, such as in a wireless sensor network [2, 3]. Android devices acting as soft access points can extend the coverage of the network while simultaneously functioning in other capacities (such as that of a sensor node), reducing the amount of infrastructure required for deployment of the network over a given area.

While being comparable in size to many microntroller-based boards, the open source and popularity of Android devices have produced an enormous number of applications and algorithm implementations. Android devices are capable of executing complex algorithms directly on board, giving the Android mesh network facility for distributed computing.

This paper demonstrates how wireless mesh networks can be constructed by Android devices automatically with Wi-Fi Direct [4] using a technique that establishes connections without requiring user interaction. By allowing these devices to connect to each other automatically, we make their use practical in a number of different applications in which they would not otherwise be, namely, ones which require them to be permanently or semi-permanently installed - as network infrastructure or as part of a larger stationary sensor network, for example. Used Android devices are easy to obtain and inexpensive; they will only become more so in the future, making them potentially cost efficient alternatives to more traditional types of infrastructure.

In section 2, we review the concept of a wireless mesh network, acknowledge existing, similar approaches, and discuss Wi-Fi Direct on Android. In Section 3, we discuss the existing network implementations for Android. Sections 4, 5, and 6 outline our implementation in detail, describing both methodology and architecture. Section 7 shows the structure of the software, and section 8 discusses our testing and results.

2 Background

2.1 About Wireless Mesh Networks

Wireless Mesh Networks are dynamic, self-organizing networks made up of devices which are either mesh clients or mesh routers. Wireless Mesh Networks are similar to ad-hoc wireless networks in structure, but differ in that for a Wireless Mesh Network, there is still a central gateway through which most network traffic will ultimately pass[5].

Wireless Mesh Networks are an efficient and cost-effective way of providing wireless network infrastructure, because of the ease and low cost of installation[5].

2.2 About Android

The key strengths of the Android platform in the context of automatically constructing Wireless Mesh Networks are:

- Android is open. The source code for the operating system is freely available, and, as a mobile platform, applications may be developed and distributed for it without any significant limitation[6].

- An application which will run on one Android device will run on another irrespective of its architecture.

- Android devices have a host of built-in sensors. Cameras, GPS, accelerometers, microphones, compasses, barometers, EM field, RGB light sensors, and gyros are all commonly available and programmer-accessible through the Android API.
- Android devices are able to communicate over 2.5-4G networks, Wi-Fi a/b/g/n, and Bluetooth®.
- Android devices are power efficient, small, and portable.
- Android devices are typically run by powerful, multi-core ARM processors, giving them the ability to perform complex tasks and computations on board.
- The market for Android devices is directly consumer driven. Development of new hardware is rapid; models of perfectly serviceable devices are succeeded almost yearly, causing the prices of the older models to drop along with the demand for them, making them a prime inexpensive target for building wireless mesh networks.
- Android applications can be configured to start automatically after device boot-up.

2.3 Wi-Fi Direct on Android

Since Android 4.0, Wi-Fi Direct has been available on Android devices through the API's Wi-Fi P2P framework. A major problem with the implementation is that, if used in any of the ways shown in the documentation or sample code[7], many-to-one connections will not work; as of Android 4.4, attempting to connect to any third device will cause the existing connection to drop. We describe the method we have used to bypass this issue in section 4.2.1.

3 Existing mesh/ad-hoc network implementations for Android

3.1 A multilayer application for multi-hop messaging on Android devices

Ryan Berti, Abinand Kishore, and Jay Huang of the University of Southern California developed software for the Android which can send data through a daisy-chain of phones linked through transient Wi-Fi Direct connections. Connections between devices are one to one, and each phone in the network acts as both a relay and a client. The application computes routes through the devices, and packs the route data in every message so that each relaying device knows where to send the data towards[8].

3.2 The serval project

The Serval Project aims to be a backup telecommunications system that can function outside a normal cellular network. The Serval Project's software can use Android's Wi-Fi in ad-hoc mode to establish its network, however, using Android's ad-hoc mode presently requires root permissions on the device (root permissions can only be acquired on a device by hacking it, which usually voids the warranty and may be illegal if the device is a contract phone). If root permissions are not available on a device, a wireless access point is needed for the device to communicate using the software[9].

3.3 Open garden

Open Garden is proprietary software for the Android, Windows, and Mac OS X that allows users to create and take part in wireless mesh networks to access the Internet. It does not require root access on a phone and can use Wi-Fi Direct or Bluetooth® to establish connections[10].

4 Establishing Android Wi-Fi Direct connections automatically

4.1 Motivation for connection automation

Android devices such as smartphones, are designed to be user-driven devices. That is, the Android stack, which sits on top of Linux, is geared towards facilitating communication, productivity, and entertainment applications in which user interaction is central. It is in this user-centric context that Android's Wi-Fi Direct framework was created.

The primary connection methods of the framework, and, indeed, the *only* documented methods for establishing peer-to-peer connectivity between Android devices are all user driven; they must have a user's input and confirmation in order to proceed with connection and authentication. While suitable for the application areas for which Android was primarily designed, the requirement of having periodic user interaction to drive connection processes limits the ways the devices can be used. Permanent installation of these devices, for example, would be impractical if a user or administrator had to periodically press a button on an authentication dialog to keep a network running.

By allowing Android devices to connect to each other automatically and without any interaction, we eliminate this requirement of user supervision and make the devices practical in applications where they may be difficult to access and/or frequent and periodic user oversight cannot or should not be guaranteed.

4.2 Implementation

In our implementation, a device in the mesh network is, at any given time, either *a*) a mesh router or *b*) a mesh client. Many clients may be connected to a single router simultaneously and can access the outside network through that router.

If necessary to enlarge coverage, multiple routers may exist in the network. If two routers are outside of each others' ranges, a client that is within the range of both can act as a relay to ferry information from one router to the other. To accomplish this, the client acting as a relay temporarily disconnects from its original mesh router to broadcast and receive data to and from the other mesh router[5].

Unlike some implementations, ours does not need root access on any device. It can also run automatically and establish connections without user intervention, and works on any device running Android 4.0 or later. It is also lightweight enough to be used for communications infrastructure.

The two main obstacles overcome by our implementation are:

1. As mentioned previously in section 2.3, as of Android 4.4 (API 19), the Wi-Fi Direct interface on Android **may not** be used directly to connect more than two devices at a time.

2. When two devices try to connect to each other using Wi-Fi Direct, **manual, user-interactive authentication is required**. A WPS confirmation dialog box will pop up on the screen during a connection attempt and must be answered by a user in order for the connection to be accepted. A device that has already connected once may be "remembered," which means that if it attempts to connect again no authentication dialog will be presented. However, **remembered devices are *forgotten* after reboots**.

4.2.1 Soft access point solution

Though the Android Wi-Fi Direct implementation does not work properly for many-to-one connections (see section 2.3) it *does*, however, allow for the creation of a soft access point (AP) that one or more devices can connect to simultaneously if they know its SSID and preshared key.

Since neither the SSID or preshared key of the access point can be set by the framework (both are automatically and randomly generated), they must somehow be given to the devices which are to connect to the soft access point.

The way we have accomplished this is by packing the SSID and preshared key into a network service discovery broadcast, which can be sent wirelessly from the soft access point to other devices in range using the Android Wi-Fi Direct framework. The SSID and the preshared key are both encrypted with a separate key that all the devices know beforehand (an encryption key, unlike an access point preshared key, can be set programmatically).

All devices within range of one of these soft access points (AP) will receive this network service discovery broadcast containing the AP's SSID and PSK (*see figure 1*). If a client device is within range of more than one soft access point, it will receive the AP information for all of them, and if it is to act as a relay, it can take turns connecting to each AP.

Using this method, multiple clients can connect to a single device over Wi-Fi simultaneously (*see figure 2*); a second and very important benefit to this method is that the authentication process can be completely automated.

4.2.2 Security

Soft access points created by Android's Wi-Fi Direct framework are WPA2 compliant, which means that all communications between the access point and authenticated clients are encrypted using AES-CCMP or TKIP[11].

The wireless AP password is encrypted inside the network service discovery broadcast with the javax.crypto.Cipher class using "AES/GCM/PKCS5PADDING", which is the AES algorithm in Galois/Counter mode with the cleartext padded into 8 byte blocks. The key used for encryption is stored on each device beforehand to enable them to create and join the p2p network without periodic user intervention.

The reason why the preshared key of the soft access point is not just directly stored beforehand is because it is randomly generated and subject to change each time the soft AP is created, at the discretion of the framework and not the programmer; there is no way to change or modify it. If the preshared key was stored beforehand, then users would have to reconfigure all the devices whenever an Android device running a soft access point decided to change its preshared key. In other words, we cannot effectively "preshare" the preshared key because it changes at the whim of Android OS.

By broadcasting the AP's most recent preshared key each time it is created, we ensure that clients can always connect to it without having to be manually reconfigured. By encrypting the key before broadcasting it, we ensure that only the clients who we want to be able to connect to the network will be able to; we can store the encryption key used to encrypt the AP keys on the devices beforehand, without having to worry about it changing.

4.3 Mesh router procedure

1. **Create a soft AP** using Wi-Fi Direct. This is performed by making a group using the *createGroup* method of the `WifiP2pManager` class.
   The newly created access point:

   (a) creates its SSID from its device name and a random string of letters.

   (b) is secured using WPA2-PSK; its passphrase is randomly generated and *cannot* currently be changed or set by the programmer.

   (c) runs a DHCP service on the device to assign local IP addresses to connecting client devices. This device is known as the "group owner"

afterwards[12].

2. **Wait for the soft AP to finish starting up.** The Android device will send a WIFI_P2P_-CONNECTION_CHANGED_ACTION intent when the AP is up. Associate a `BroadcastReceiver` with this intent to catch it. This BroadcastReceiver will be called by the operating system:

   (a) whenever a change in the Wi-Fi Direct connection is detected.

   (b) after *createGroup* finishes creating the access point.

   Receipt of this intent will act as a signal to proceed to the next step.

3. **Pack the SSID, passphrase, MAC, and connectivity status into a string.**

4. **Encrypt the string.** Use an encoding for this string that will not lose information when cast into Unicode, such as ISO-8859-1[13].

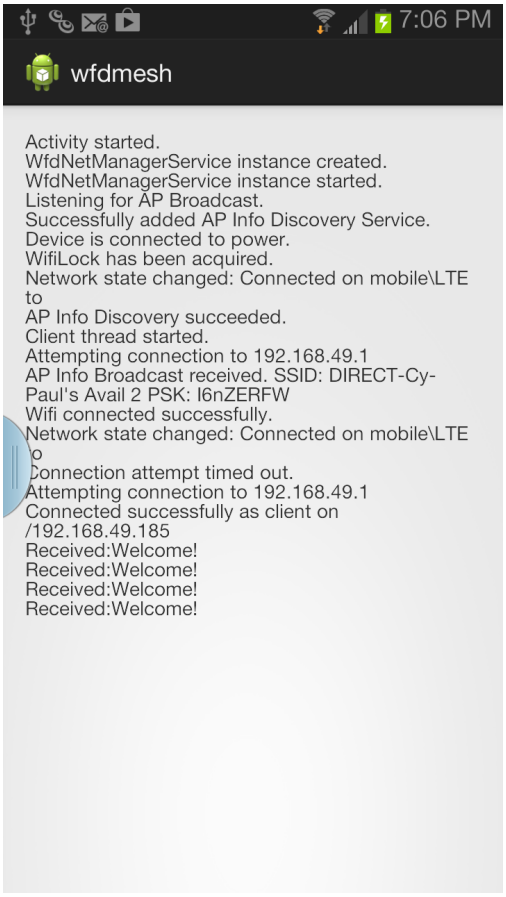


Figure 1: A client device picking up an AP broadcast and connecting.

5. **Create a `WifiP2pDnsSdServiceInfo` object** and put the encrypted string into it. This can be done from inside the `BroadcastReceiver`.

6. **Begin broadcasting the service information** by calling WifiP2pManager's *addLocalService* method.

7. **Start a netsock server** on a separate thread and begin listening for clients attempting to connect. In our implementation, each connected client is given its own new thread for the sake of simplicity.

Router Pseudocode:

```
CreateAP()
en_ssid ← Encrypt(GetMySSID())
en_psk ← Encrypt(GetMyPSK())
MakeNSDBroadcast(en_ssid, en_psk)
server_socket ← OpenServerSocket()

while program_is_running
  client_socket ← BlockingAccept()
  StartNewClientIOThread(client_socket)
```

4.4 Client procedure

1. **Make a callback object to receive network service discovery (NSD) broadcasts** by creating a `DnsSdServiceResponseListener`. This will be called any time a Upnp broadcast is received. The `DnsSdServiceResponseListener` must:

   (a) Receive the broadcast data as a String

   (b) Unencrypt the received broadcast string

   (c) Parse out the SSID and passphrase from the unencrypted string

   (d) Store the parsed out information in an instance of a Class designed to hold it

   (e) Push that instance with the data into a list.

2. **Start listening for network service discovery broadcasts** by calling WifiP2pManager's *setDnsSdResponseListeners*, *addServiceRequest*, and *discoverServices* methods to register, associate, then start the UpnpService discovery service with the `DnsSdServiceResponseListener`.

3. **Begin a timer that will stop the NSD listening some time after the receipt of the first broadcast.** This gives the network service discovery broadcast receiver time to pick up other broadcasts, if there are any.

4. **Decide which access point among those heard from is the best to connect to** once the time expires. The "best" may be the first AP in the list that is connected to the outside network, the AP with the strongest radio signal, or the only AP available.

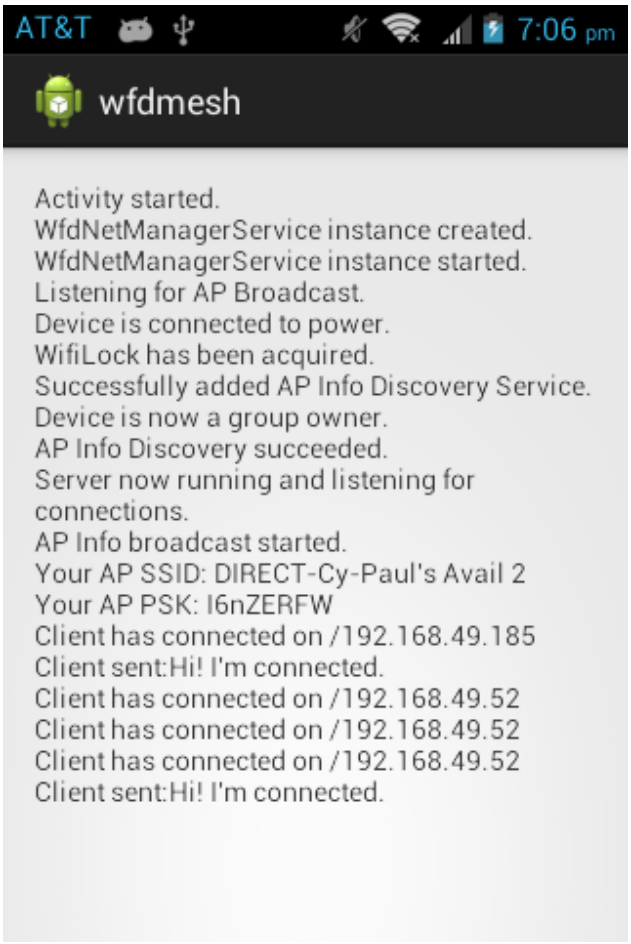


Figure 2: Server broadcasting AP info and allowing multiple clients to connect simultaneously.

5. **Connect to the "best" AP** using WifiManager.

6. Once the device has connected to the mesh router over Wi-Fi, **connect to the router's netsock server** and begin exchanging information.

Client Pseudocode:

$x \leftarrow 0$
while *listening_for_NSD_broadcasts*
  if *received_new_broadcast*
    $AP[x].SSID \leftarrow$ `Decrypt`(Broadcast.SSID)
    $AP[x].PSK \leftarrow$ `Decrypt`(Broadcast.PSK)
    $x \leftarrow x + 1$
$maxaps \leftarrow x$
$i \leftarrow 1$
$t \leftarrow$ *arbitrary time interval*
$server\_ip\_address \leftarrow$ "192.168.49.1"

while *program_is_running*
  `ConnectToAP`($AP[i]$)
  $csocket \leftarrow$ `ConnectSock`($server\_ip\_address$)
  `PerformIO`($csocket$)
  `wait`(t)
  `DisconnectFromAP`()
  if $i \leq maxaps$
    $i \leftarrow i + 1$
  else $i \leftarrow 0$

4.5 The relay procedure

Clients which are to act as relays between access points remember all the access points they heard from during the initial broadcast listening period. For now, they connect to all the access points in range in a round robin fashion, switching from one access point to the next after an arbitrary amount of time (*see figure 3*); our future work will explore more sophisticated routing schemes, such as demand and event-based relaying.

1. After connecting to the first access point in its list and receiving data, the relay client will disconnect from the AP.

2. After disconnecting, the relay client will connect to the next AP in its list.

3. While connected to this AP, the relay client will broadcast any information it knows that it needs to forward. The mesh router will have timestamped all messages it has queued as outgoing, and will check these timestamps against the last time this client disconnected. Messages which are more recent than this time will be sent to the client.

4. After all data is transmitted, upon instruction from its access point, or after an arbitrary amount of time, the client relay will disconnect from this access point device and connect to the next one (which is the first access point in its list if it has reached the end of the list).

5 Broadcasting a message

1. The message to be sent is prefixed with a universally unique identifier (UUID) used to identify it, and the MAC address of its origin. The UUID is randomly generated when the message is created and is immutable.

2. If a router or a node receives a message with a UUID that it has already received recently, the message is ignored and discarded, not to be passed on. An arbitrary number of UUIDs may be stored in a circular buffer from which to check against. The ideal size of the buffer will depend on the size of the network and the frequency of data exchange.

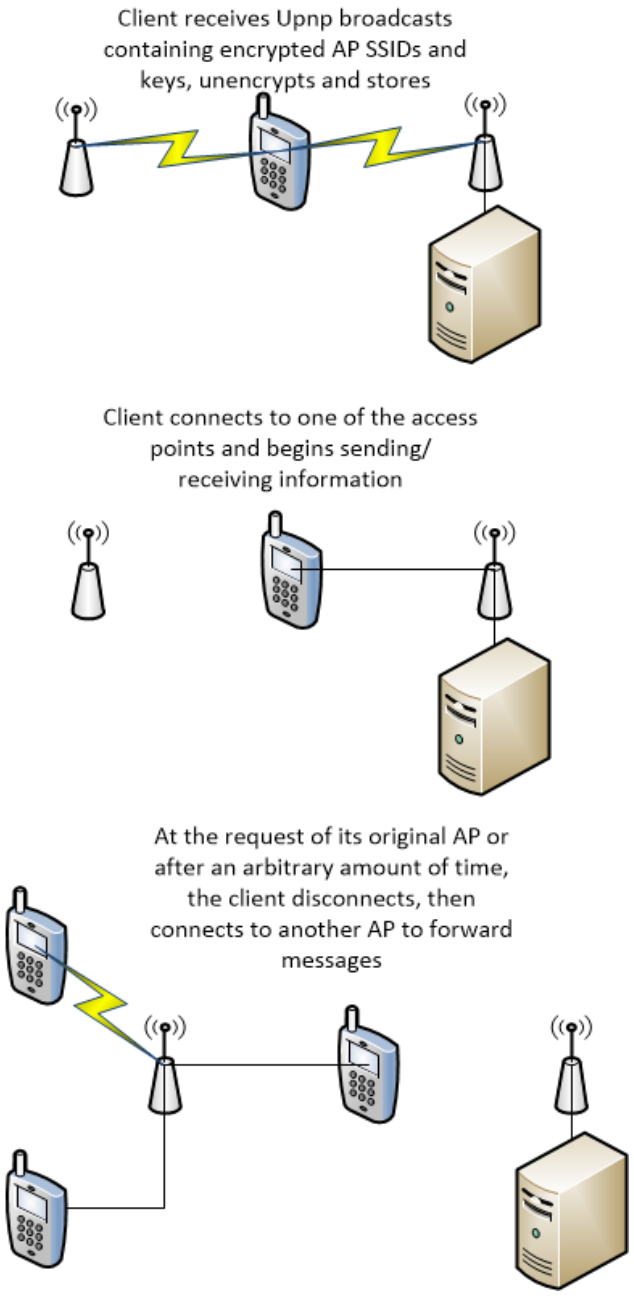


Figure 3: Relay procedure.

3. If the message is new, process the information, append the device's MAC to the header, then send the message out.

5.1 Message structure

The message may contain either data or commands, and may be, as described in the previous sections, broadcast to all devices or sent to a specific device only.

The parts of the message are (*see figure 4*):

1. **Header**:
   (a) *UUID:* Uniquely identifies a message. Useful in determining whether or not a message has been received before.
   (b) *Message Type Data:* Indicates whether the message contains one of the following:
      – data intended for a single recipient
      – data intended for all devices on the network
      – a command intended for a single recipient
      – a command intended for all devices on the network
   (c) *MAC address list:* Used to construct route data and to determine the directions the message is to be passed in (Note: the first two characters of a MAC address may change for the same device and same network interface, and should be ignored.)
2. **Data**: The payload of the message. May be anything from a command to raw binary data.

5.2 Sending a message intended for a single recipient

If the network is to send messages meant for particular devices, the steps for broadcasting a message may be used to construct a graph of the network. After step 5, you would check to see if the recipient of the message can connect to any devices other than the mesh router to which it is already connected. If it cannot, the message has reached a "leaf" node, and sends the data back up to the source of the message. The header will contain, in order, the MAC addresses of each mesh router and mesh client which the message has passed over on its way to the leaf.

Depending on the needs of the network, it may be simpler to broadcast all messages, regardless of whether they are intended for a single recipient. The devices can all check to see if the message was intended for them, and ignore its contents if it was not (though still passing it along as needed). In this fashion, the message will eventually reach the device it is supposed to, though it may not take the shortest path.

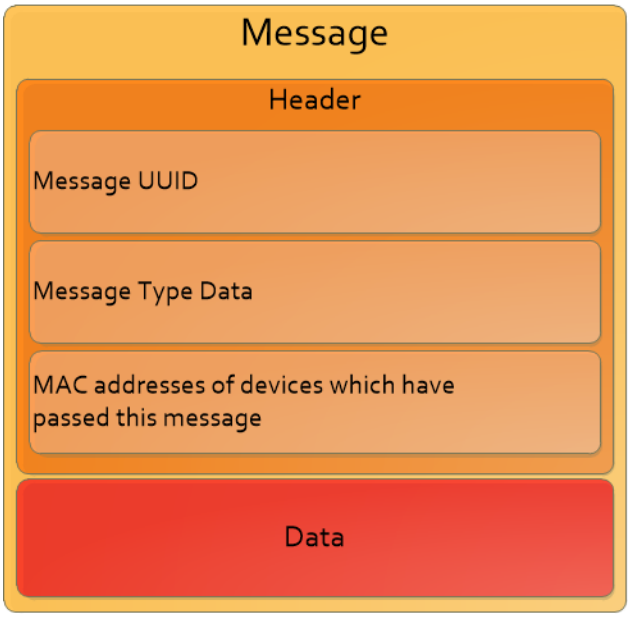


Figure 4: Internal structure of a message.

6 Starting the software on device boot

An application can be configured to start after the Android OS has completed booting. Permission for `android.permission.RECEIVE_BOOT_COMPLETED` must be requested in the application's manifest file, and a receiver for the `android.intent.action.BOOT_COMPLETED` intent must be declared. The OS will start the application after it is completed booting, then send the specified receiver this intent, after which the receiver can start up the main activity.

Also, the lock screen can be disabled manually in the Security section of the device's settings; by default, it will be shown as soon as the device powers on.

7 The structure of the software

Our software can be installed on any Wi-Fi capable device running Android 4.0 or later, and can be divided into three main parts (*See figure 5*):

1. The main activity, which provides a console for on-device logging and debugging, as well as basic UI functionality
2. The network service, which either runs the mesh router service or the mesh client service
3. The `BOOT_COMPLETED` broadcast receiver, which is registered with the system from the application manifest, and the starting point of the program if started by the system

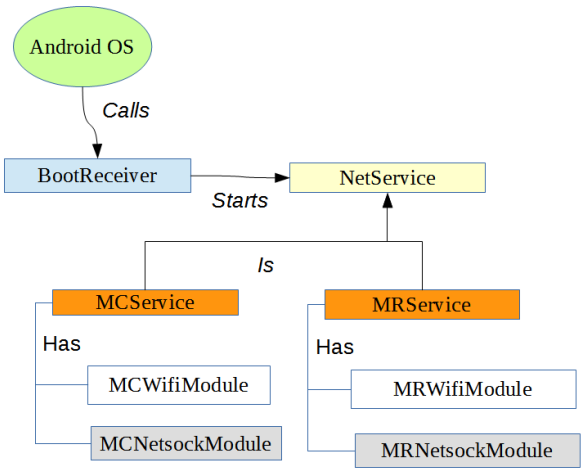


Figure 5: The overall architecture of the software.

7.1 Main activity / boot receiver

The first time the program is run on a device, the main activity will ask whether the device is to be a mesh client or a mesh router via a dialog box. This only needs to be done once, as the choice will be saved to file and remembered the next time the program starts.

The program can be manually set to either router or client mode at any time from an action bar menu.

After being configured for the first time, the program will run automatically on device boot through its Boot Receiver, which is called by the operating system.

7.2 Network service

The network service started by the main activity is either the mesh router service or the mesh client service, both of which inherit from a common "network service" superclass and present an identical I/O interface, which can be used to query the service if it can talk to any other device, as well as send and listen for messages.

The services themselves are divided into two modules:

1. the Wi-Fi Direct and Wi-Fi module, which is responsible for *a*) advertising and listening for network service discovery broadcasts containing access point information, and *b*) establishing and maintaining Wi-Fi connections.
2. the netsock module, which is *a*) started by the previous module after a Wi-Fi connection has been established, *b*) transfers data over TCP sockets, and *c*) is multi-threaded for efficiency, and to keep the main activity thread from being killed by the operating system.

7.3 Attaching other components

Our software provides network communications, and may be either used as part of other software, or potentially run on its own and used through Android's IPC mechanisms. Messages may arrive asynchronously depending on a device's location in the mesh network; this must be taken into account by the code to use this service.

8 Testing and results

Our build environment was Eclipse with the Android Developer Tools plugin running on both Windows and Linux. The devices we deployed our software on included a pair of twin Samsung Galaxy Nexuses, the Motorola Moto, and the Nexus 7, which used Android versions 4.3, 4.4.2, and 4.4.3, respectively. All of the phones were running either OEM versions of Android, or locked, contract-phone versions of it. The software is written entirely in Java using the Google Android APIs and does not require any of the devices to be rooted.

During testing, we were able to establish the network on up to four devices, with one acting as a soft AP tethered to the Internet, another acting as a soft AP with no direct Internet connection, a client acting as a relay between the two, and a client persistently connected to the same soft AP. We were also able to configure the devices so that three of them acted as soft APs, while the last worked as a relay between all of them. In both configurations, the phones were able to construct the network automatically after being turned on. One of the challenges we have experienced is making a device's NSD broadcasts continuously visible to other devices. Sometimes, this required us to power-cycle the wi-fi radio, clear the phone's program cache, or restart the device several times.

With a clear, unobstructed line of sight, the wireless radios on the Android devices we tested were capable of maintaining a direct connection to a device-created soft access point at distances up to 32 to 35 meters. We found that obstructions such as walls, doors, etc. *significantly* reduced the operable range.

9 Conclusion

In this article, we discussed and demonstrated how the Android's Wi-Fi Direct API can be used to automatically establish connections to other Wi-Fi Direct devices. By using network service discovery broadcasts over Wi-Fi Direct, we are able to transmit a soft AP's preshared key to other devices, then connect these devices to the AP using their regular Wi-Fi facilities without ever requiring a user to perform manual WPS authentication on any of them.

Also, we have shown that, once freed from the requirement of manual user authentication, it becomes practical for Android devices to automatically construct and participate in a wireless mesh network, acting as mesh routers, mesh clients, and relays.

10 Acknowledgements

We thank Dr. Robert Ryan and Mary Pagnutti from Innovative Imaging and Research for their advice, guidance, and feedback on this paper. The material in this paper is based on work supported by NASA STTR Phase 2 contract No. NNX13CS14C.

References

[1] Google, "Android," Jan. 2014. [Online]. Available: http://www.android.com/

[2] W.-J. Yi, W. Jia, and J. Saniie, "Mobile Sensor Data Collector using Android Smartphone," *Circuits and Systems (MWSCAS), 2012 IEEE 55th International Midwest Symposium on*, no. 55, pp. 956–959, 2012.

[3] P. P. Jayaraman, A. Zaslavsky, and J. Delsing, "Sensor Data Collection Using Heterogeneous Mobile Devices," *Pervasive Services, IEEE International Conference on*, pp. 161–164, 2007.

[4] "Wi-fi direct," Jan. 2014. [Online]. Available: http://www.wi-fi.org/discover-wi-fi/wi-fi-direct

[5] *Wireless Mesh Networks: Architectures and Protocols*. Springer, 2007.

[6] R. Meier, *Professional Android 4 Application Development*. Wrox, 2012.

[7] Google, "Wifip2pmanager," Jan. 2014. [Online]. Available: http://developer.android.com/guide/topics/connectivity/wifip2p.html

[8] K. Huang, Berti, "A multilayer application for multi-hop messaging on android devices," Oct. 2013. [Online]. Available: http://anrg.usc.edu/ee579_2012/Group02/Design.html

[9] "Serval mesh documentation," Oct. 2013. [Online]. Available: http://developer.servalproject.org/

[10] "Opengarden.com," Oct. 2013. [Online]. Available: https://opengarden.com/

[11] "IEEE 802.11i-2004: Part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications," Jul. 2004.

[12] D. Camps-Mur, A. Garcia-Saavedra, and P. Serrano, "Device-to-Device Communications with WiFi Direct: Overview and Experimentation," *IEEE Wireless Communications*, vol. 20, no. 3, pp. 96–104, 2013.

[13] "ISO-8859-1: Information technology – 8-bit single-byte coded graphic character sets – Part 1: Latin alphabet No. 1," 2003.

Visualization and Concept Drift Detection Using Explanations of Incremental Models

Jaka Demšar, Zoran Bosnić and Igor Kononenko
University of Ljubljana, Faculty of Computer and Information Science
Večna pot 113, 1000 Ljubljana, Slovenia
E-mail: jaka.demsar0@gmail.com, {zoran.bosnic, igor.kononenko}@fri.uni-lj.si

Keywords: data stream mining, concept drift detection, visual perception

Received: October 10, 2014

The temporal dimension that is ever more prevalent in data makes data stream mining (incremental learning) an important field of machine learning. In addition to accurate predictions, explanations of the models and examples are a crucial component as they provide insight into model's decision and lessen its black box nature, thus increasing the user's trust. Proper visual representation of data is also very relevant to user's understanding – visualization is often utilised in machine learning since it shifts the balance between perception and cognition to take fuller advantage of the brain's abilities. In this paper we review visualisation in incremental setting and devise an improved version of an existing visualisation of explanations of incremental models. Additionally, we discuss the detection of concept drift in data streams and experiment with a novel detection method that uses the stream of model's explanations to determine the places of change in the data domain.

Povzetek: V članku predstavimo novo vizualizacijo razlage inkrementalnih klasifikacijskih modelov in posameznih napovedi. Predlagamo tudi metodo zaznave spremembe koncepta, temelječo na nadzorovanju toka razlag.

1 Introduction

Data streams are becoming ubiquitous. This is a consequence of the increasing number of automatic data feeds, sensoric networks and internet of things. The defining characteristics of data streams are their transient dynamic nature and temporal component. In contrast with static (tabular) datasets (used in *batch* learning), data streams (used in *incremental* learning) can be large, semi-structured, incomplete, irregular, distributed and possibly unlimited. This poses a challenge for storage and processing which should be done in constant time – for incremental models, operations of *increment* (fast update of the model with the new example) and *decrement* (forgetting old examples) are vital. Concepts and patterns in data domain can change (*concept drift*) – we need to adapt to this phenomenon or the quality of our predictions deteriorates. We discuss data streams and concept drift more in Section 2.1.

Bare prediction quality is not a sufficient property of a good machine learning algorithm. *Explanation* of individual predictions and model as a whole is needed to increase the user's trust in the decision and provide insight in the workings of the model, which can significantly increase the model's credibility. The model independent methods of explanation have been developed for batch [12, 13] and incremental learning [2] (Section 2.2).

Data visualisation is a versatile tool in machine learning that serves for sense-making and communication as it conveys abstract concepts in a form, understandable to humans. In Section 2.3 we discuss visual display of data in an incremental setting and describe the improper visualisation of explanation of incremental models. The main goal of the article is its improvement, which is presented in Section 3. An additional goal (Section 4) was to devise a method of concept drift detection which monitors the stream of explanations. Finally, we test the improved visualization and the novel concept drift detection method on two datasets and evaluate the results (Section 5).

2 Related work

2.1 Data stream mining

In incremental learning, we can observe possibly infinite data stream of pairs (x_i, C_i), where x_i is the $i-th$ instance and C_i is its true label. After the model makes a prediction $p_i = \hat{C}_i$, the environment reacts with a feedback which can be used to assess the model's performance (in the case of classifiers, the instance's true label becomes available). According to *PAC* (*Probably approximately correct*) learning model, if the distribution, generating the instances is stationary, the error rate will, at least for sound machine learning algorithms, decline towards the Bayes error rate as the number of processed instances increases [10]. Consequently, when a statistically significant rise in error rate

is detected, we can suggest that there has been a change in the generating distribution – concept drift. The nature of change is described with two dimensions: the cause of change (changes in data, hidden variables, changes in the context of learning...) and the rate of change. The change detection can be also completely left out (using windows and blindly forcing the operations of increment and decrement). The other approach is to actively detect change either by monitoring the evolution of performance measures or comparing distributions over two time windows. In the explanation methodology (Sections 2.2 and 4) we use two methods from the former group.

The basis of monitoring the learning process using *statistical process control* (*SPC*) [4] is detecting significant error rate using central limit theorem. Each processed instance is in one of the three possible states - *in control*, *out of control* and *warning* when the system is in between the former states. When *in control*, the current model is incrementally updated with the current instance, since the error rate is stable. In *warning* state, a buffer is filled with incoming instances – it serves as an optimally-sized container for data that is relevant for the new model if the drift occurs (*out of control* state) – the buffer is then used to construct a new learning model form the instances in it. If the system goes from *warning* back to *in control*, the buffer is emptied, since we deemed the error rise to be a false alarm.

The other method, *Page-Hinkley test* [8], is simpler in its nature. It was devised to detect the change of a Gaussian signal and is commonly used in signal processing. The method's behaviour can be controlled with parameters λ (threshold for alarm) and δ, which corresponds to the allowed magnitude of changes.

2.2 Explanation of models and individual predictions

Explanation of individual predictions and prediction models is used to gain additional insight into the model's decision and to lessen the black box nature of most predictions. The adequately explained prediction increases the user's trust and understanding of the model and extracts additional information.

Although some models are already transparent (e.g. decision trees, Bayesian networks) and model-dependent methods of explanation exist for most others, these methods do not meet the requirement for *consistency of explanation*, which enables comparison of explanations across different models. *IME* (*Interactions-based Method for Explanation*) [13] with its efficient adaptation [12] is a model independent method of explanation, which also addresses interactions of features and therefore successfully tackles the problem of redundant and disjunctive concepts in data.

The explanation of the prediction p_i for instance x_i is defined as a vector of *contributions of individual feature values* $(\varphi_1, \varphi_2, \ldots, \varphi_n)$ where φ_j is the contribution of the value of the j-th feature. Positive φ_j implies that the feature value positively contributed to the prediction (and vice versa) while the absolute value $|\varphi_j|$ is proportional to the magnitude of influence on the decision. The sum of all contributions is equal to the difference between the prediction using all feature values and a prediction using no features (*prediction difference*). The explanation quality in highly correlated with prediction quality [13]. Consequently, for very good models, the explanation gives us an insight not only into the working of the built model, but also into concepts behind the domain itself.

In order to discover dependencies among feature values, 2^n subsets of features are considered. An efficient sampling-based approximation method exists [12], where the explanation of a prediction is modeled with game theory as a cooperative game (N, Δ) where players N are features and Δ is the prediction difference. The payoff vector divides the prediction difference among feature values as contributions that correctly explain the prediction[1]. The explanation of a single prediction can be expanded to the whole model by iterating thorough all possible feature values and using contributions of randomly generated instance pairs to compute the contributions of each attribute value.

An adaptation of existing explanation methodology to incremental setting is proposed in [2]. In addition to the efficiency requirements, the key feature of incremental learning to consider is the possibility of a concept drift. The explanation in incremental learning should therefore itself be a data stream. An adaptation is developed by considering the existing model-independent method in batch learning [12] and equipping it with drift detection and adaptation methods.

In the incremental explanation methodology, the basic concept is slightly modified by introducing the parameter max_window. It acts as a limiter for maximum size of the learning model, by narrowing down the model by FIFO principle, if necessary. The batch explanation is integrated at key points in SPC[2] (changes of states), resulting in a granular stream of explanations which reflect local areas of static distributions and intermediate areas of concept drift. An optional parameter ω determines triggering of periodic explanations independent of change indicators. Aside from periodic explanations, the granularity of the stream is completely correlated with change detection – the explanation of the model occurs only when error rate significantly increases, indicating a change in the distribution of generating the instances[3]. Explaining an individual prediction follows the same process as in batch explanation, only that the local learning model is used.

[1] The concept of Shapley value is used as a solution.

[2] SPC meets the requirement of model-independence – it works as a wrapper for an arbitrary classifier. In addition to the occurrence of a concept drift, it also detects its rate (the smaller the buffer is between *warning* and *out of control*, the faster the drift occurs)

[3] The resulting stream of explanations itself can be a subject of analysis – if the model is good, patterns found in the explanation stream may reflect the patterns behind the domain.

2.3 Data visualisation

Data visualization is often utilised in machine learning since it shifts the balance between perception and cognition to take fuller advantage of the brain's abilities [3]. A data stream can be seen as a collection of observations made sequentially in time. Sampling based research shows that the majority of visualizations depict data that has a temporal component [9]. In this context, visualization acts as a form of *summarization*. The challenge lies in representing the temporal component, especially if we are limited to two-dimensional non-interactive visualisations. Concept drift is the other property in data streams that makes creating good visualization a hard task. Even if we manage to effectively summarize and display patterns in data at some point, we are still left with the task of displaying the change in them.

The main goal of this paper is improving the existing methodology for visualising explanations of incremental models [2]. The feature value contributions are represented with customised bar charts. When explaining a static model, all possible values are listed along the axis with mean negative and mean positive contributions of each feature. When plotting multiple such visualisations (as is the case in locally explaining incremental models based on change detection) they become very difficult to read as a whole because of the large number of visual elements that we have to compare (we sacrifice macro view completely in favour of micro view). To consolidate these images and address the *change blindness* [7] phenomenon, charts are stacked into a single plot, where the age and size of the explanation are represented with transparency (older and "smaller" explanations fade out). The resulting visualisation is not tainted by first impressions (as it is only one image) and is adequately dense and graphically rich. However, the major flaw of this approach lies in the situations when columns, representing newer explanations override older ones and thus obfuscate the true flow of changing explanations, for example, when the concept drift precipitates the attribute value contributions to increase in size without changing the sign. Concepts can therefore become not only hidden; what's more, the visualization can be deceiving, which we consider to be worse than just being too sparse. Therefore, we need to clarify the presentation of the concept drift along with an accurate depiction of each explanation's contributions while maintaining the macro visual value, that enables us to detect patterns and get a sense of true concepts and flow of changes behind the model.

3 Improved visualisation

When visualising explanations of individual predictions, horizontal bar charts are a fitting method also in the incremental setting – we plot the mean positive and negative contribution of each attribute value and the mean of each attribute as a whole. Individual examples are always explained according to the current model which, in our case, can change. This is not an obstacle, since the snapshot of the current model is in fact the model that classified the example, so we can proceed with the same explanation methodology as in the static environment[4].

This approach fails with explanations of incremental models as we need a new figure for each local explanation (Subsection 2.3). To successfully represent the temporal component of incremental models, we use two variations of a line plot where the x axis contains time stamps of examples and the splines plotted are various representations of contributions (y axis).

The first type of visualization (examples in Figures 2 and 4) has one line plot for each attribute. Contributions of values of the individual attribute are represented with line styles. The mean positive and mean negative contribution of the attribute as a whole are represented with two thick faded lines. Solid vertical lines indicate the spots where explanation of the model was triggered (and therefore become the joints for the plotted splines), while dashed vertical lines mark the places where the actual concept drift occurs in data. The second type is an aggregated version (examples in Figures 3 and 5) where the mean positive and mean negative contributions of all attributes are visualized in one figure. In these two ways we condense the visualization of incremental models without a significant loss in information while still providing a quality insight into the model. Exact values of contributions along with timestamps of changes can be read out (micro view), while general patterns and trends can be recognised in the shapes of lines that are intuitive representations of flowing time (macro view). The resulting visualisations are dense with information, easily understandable (conventional plotting of independent variable, time, on x-axis) and presented in gray-scale palette, making them more suitable for print.

4 Detecting concept drift using the stream of explanations

When explaining incremental models, the resulting explanations are, in themselves, a data stream. This gives us the option to process them with all the methods used in incremental learning. In our case, we'll devise a method to detect outliers in the stream of explanations and declare such points as places of concept drift. The reasoning behind this is the notion that if the model does not change, then also the explanation of the whole model will not change. When an outlier is detected, we consider this to be an indicator of a significant change in the model and thus also in the underlying data. In addition to this, the method provides us with a stream of explanations that is continuous to a certain degree of granularity and so enables us to overview the concepts behind the data at more frequent intervals than the existing explanation methodology.

[4]That means that, without major modifications, we can only explain instances at the time of their classification – the model changes with the arrival of the next instance.

We use a standard incremental learning algorithm [4] (we learn by incrementally updating the model with each new example, decrement the model if it becomes too big or rebuild the model if we detect change [5]) and introduce the $granularity$ parameter which determines how often the explanation of the current model will be triggered. This will generate a stream of explanations (vectors of feature value contributions) that will be compared using cosine distance. For each new explanation, the average cosine distance from all other explanations that are in the current model, is calculated. These values are monitored using the Page Hinkley test. When an alarm is triggered, it means that the current average cosine distance from other explanations has risen significantly, which we interpret as a change in data domain – concept drift. The last $granulation$ examples are then used to rebuild the model, the Page Hinkley statistic and the local explanation storage are reset (to monitor the new model).

The cosine distance is chosen because, in the case of explanations, we consider the direction of the vector of contributions to be more important than its size, which is very influential in the traditional Minkowski distances. It also carries an intuitive meaning. The Page Hinkley test is used in favour of SPC because of its superior drift detection times [10] and the lack of need for a buffer – examples are already buffered according to the granularity. The method is therefore model independent. Algorithm 1 describes the process in a high level pseudocode.

Algorithm 1 Detecting concept drift using the stream of explanations

Require: h {classifier}
Require: $(\vec{x_i}, y_i)_t$ {data stream}
Require: m {number of samples for *IME*}
Require: g {explanation granulation}
Require: max_window {maximum size of the model}
Require: λ {Page Hinkley threshold}
  $local\_explanations \leftarrow []$
  $\Phi = h$ {Incremental classifier}
  $buf = []$ {buffer}
  for $(\vec{x_i}, y_i) \in (\vec{x_i}, y_i)_t$ **do**
    $\Phi = \Phi.increment(\vec{x_i})$
    if $len(\Phi) > max\_window$ **then**
      $\Phi = \Phi.decrement()$
    end if
    $buf.append((\vec{x_i}, y_i))$
    if $len(buf) > g$ **then**
      $buf = buf[1:]$ {Maintain buffer size}
    end if
    if $i\%g == 0$ **then**
      $\phi_i = IME(\Phi)$ {Explain the current model}
      $dist\_\phi_i \leftarrow mean(cos\_dist(\phi_i, \phi')$
        $\forall \phi' \in local\_explanations)$
      if $PageHiknley(dist\_\phi_i) = ALERT$ **then**
        $\Phi = h(buf)$ {Rebuild}
        $local\_explanations \leftarrow []$
      end if
      $local\_explanations \leftarrow local\_explanations \cup \phi$
    end if
  end for

5 Results

5.1 Testing methodology and datasets

We test the novel visualisation method and the concept drift detection method on two synthetic datasets, both containing multiple concepts with various degrees of drift between them. These datasets are also used in previous work [2], so a direct assessment of visualization quality and drift detection performance can be made. From the results of this testing, we may conclude whether the new methods are the improvements. The naive Bayes classifier and the k nearest neighbour classifier are used. Their usage yields very similar results in all tests, so only results obtained by testing with Naive Bayes are presented.

SEA concepts [11] is a data stream comprising 60000 instances with continuous numeric features $x_i \in [0, 10]$, where $i = 1, 2, 3$. x_1 and x_2 are relevant features that determine the target concept with $x_2 + x_3 \leq \beta$ where threshold $\beta \in \{7, 8, 9, 9.5\}$. Points are arranged into four blocks: $\beta = 8, \beta = 9, \beta = 7$ and $\beta = 9.5$, consecutively. Although the changes between the generated concepts are abrupt, 10% class noise is inserted into each block.

The second dataset, *STAGGER*, is generated with *MOA (Massive Online Analysis)* data mining software [1]. The instances represent geometrical shapes which are in the feature space described by discrete features *size*, *color* and *shape*. The binary class variable is determined by one of the three target concepts ($(size = small) \wedge (color = red)$, $(color = green) \vee (shape = square)$ and $(size = medium) \vee (size = large)$). We generate 4500 instances which are divided into blocks belonging to the particular concept (presented in Table 5.1). The concept drift is applied by specifying an interval of certain length between blocks – there, the target concepts of the instances mix according to the sigmoid function. Therefore, the dataset includes gradual drift, disjunction in concepts and redundant features.

Interval	Concept	Width of drift
$[0, 749]$	1	50
$[750, 1799]$	2	50
$[1800, 3599]$	3	150
$[3600, 4500]$	1	/

Table 1: STAGGER dataset used for evaluation

5.2 Improved visualizations

Concept drifts in STAGGER dataset (we use $max\_length = \omega = 500$) are correctly detected and adapted to as reflected in Figure 2. The defined concepts can be easily recognized from explanations triggered by

the SPC algorithm – the change in explanation follows the change in concept. Windows generated by the vertical lines give us insight in local explanations of the model (where the concept is deemed to be constant). Disjunct concepts (2 and 3) and redundant feature values are all explained correctly (e.g. reduncacy of "shape" and disjunction of "size" values in concept 3). Figure 1 demonstrates how classifications of two instances with same feature values can be explained completely differently at different times – adapting to change is crucial in incremental setting. This is also evident in the aggregated visualization (Figure 3), which can be used to quickly determine the importance of each attribute.

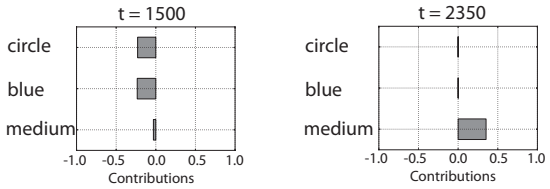


Figure 1: Explanations of individual predictions.

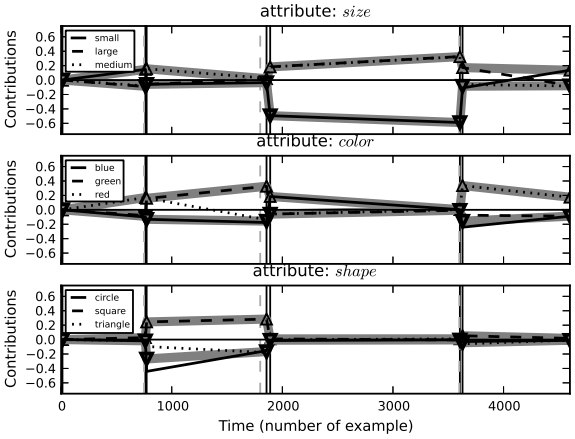


Figure 2: Visualization of explanations triggered at change detection (STAGGER).

For SEA dataset, feature values were obtained by equidistant discretization in 10 intervals for each feature. Since the blocks containing each concept are rather long and all concept drifts were successfully detected by SPC, the explanations obtained by change detection and those that were triggered periodically, do not differ greatly (we use $max\_length = 4000, \omega = 5000$). Explanations of individual instances are tightly corresponding to explanations of the model.

As can be seen in figure 4, the shape of contributions of features x_2 and x_3 reflects the target concept $x_2 + x_3 \leq \beta$; lower values increase the likelihood of positive classification and vice versa. Feature x_1 is correctly explained as

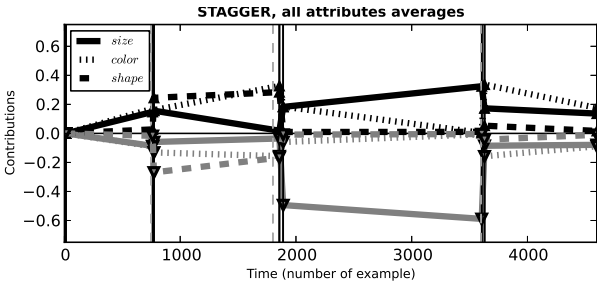


Figure 3: Aggregated visualization of explanations triggered at change detection (STAGGER).

irrelevant with its only contributions being the result of noise. The succession of target concepts $\beta \in \{8, 9, 7, 9.5\}$ is even more recognizable in the aggregated visualization (Figure 5). Changes in data are not as significant as those in STAGGER dataset, although the drift can still be observed, the dip around $t = 40000$ being a notable example (the concept drifts from $\beta = 7$ to $\beta = 9.5$).

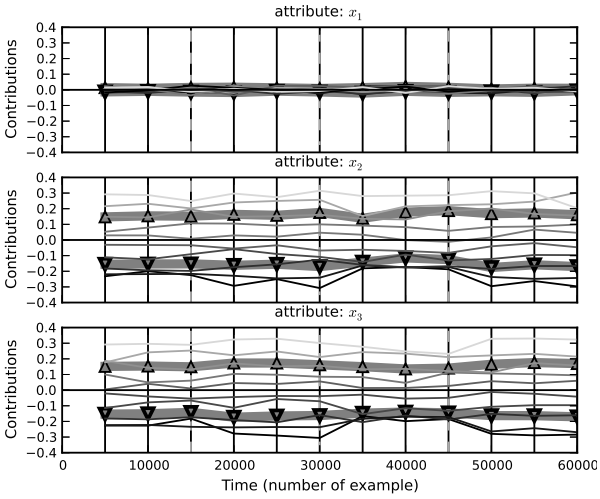


Figure 4: Visualization of periodic explanations (SEA).

5.3 Concept drift detection

Evaluating the concept drift detection using the stream of explanations on the STAGGER dataset yielded positive results. As depicted in Figure 6, the method correctly detects concept drifts without false alarms and is in that regard similar to SPC method. The stream of explanations itself fits patterns seen in testing with other successful drift detection methods (figure 2). Choices of larger granulations yielded similar results, but the change detection was obviously delayed. The concept drift was however never missed, provided that the granulation was smaller that the spacing between sequential changes in data. The delays of concept drift detection are correlated with the magnitude of change, e.g. the last concept drift ($t = 3600$) was detected

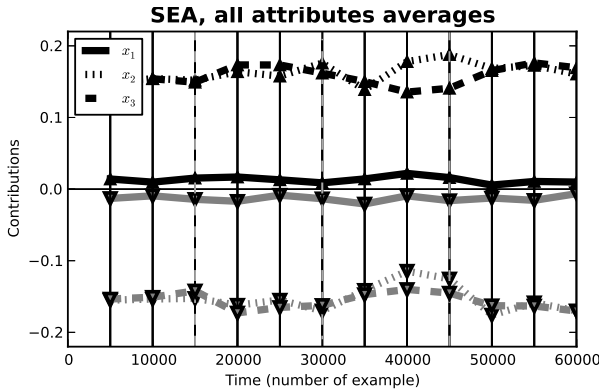


Figure 5: Aggregated visualization of periodic explanations (SEA).

with significant delay. In this regard the proposed method is inferior to SPC algorithm – the concept drift detection in noticeably delayed and we're also dependant on two parameters – granulation and alert threshold, so the generality of the method is diminished.

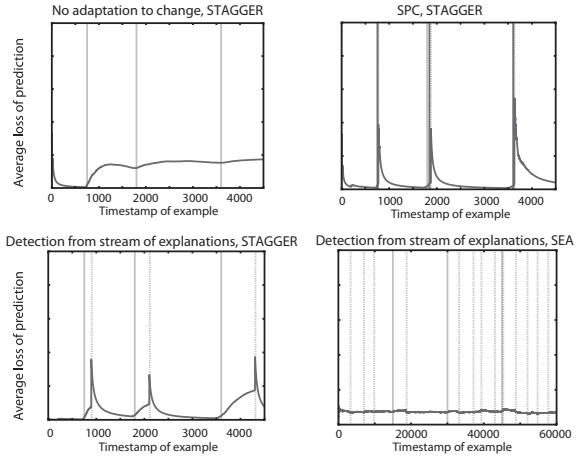


Figure 6: Loss of prediction and concept drift detection with several methods – no drift detection (static classifier), drift detection with SPC [2] and drift detection using the stream of explanations. The thick vertical lines indicate occurrence of true change in data and the dashed vertical lines mark drift detection.

When testing with SEA datasets, the concept drift was not correctly detected. Changing the granulation, Page Hinkley alert threshold and max_window parameter resulted in varying degrees of false alarms or non reaction to change (see Figure 6). This behaviour can be attributed to a small magnitude of change that occurs in data – the difference between concepts in data is quite small and continuous. However, when explaining this (incorrectly adapted) model we still recognise true underlying concepts due to the max_window property which causes the model to automatically decrement when it becomes too big[5].

We conclude that, in this form, the presented method is not a viable alternative to the existing concept drift detection methods. Its downsides include high level of parametrization (maximum size of the model, granularity and alert threshold level) which require a significant amount of prior knowledge and can also become improper if the model changes drastically. Consequently, another assessment of data is needed – the required manual supervision and lack of adaptability in this regard can be very costly and against the requirements of a good incremental model.

The concept drift detection is also not satisfactory – it is delayed in the best case or concepts can be missed or falsely alerted in the worst case. Another downside is the time complexity – the higher the granularity the more frequent explanations will be, which will provide us with a good stream of explanations but be very costly time-wise. The method is therefore not feasible in environments where quick incremental operations are vital. However, if we can afford such delays, we get a granular stream of explanations which gives us insight into the model for roughly any given time.

A note at the end: we should always remember that we are explaining the models and not the concepts behind the model. Only if the model performs well, we can claim that our explanations truly reflect the data domain. This can be tricky in incremental learning, as at the time of a concept drift, the quality of the model deteriorates.

6 Conclusion

The new visualization of explanation of incremental model is indeed an improvement compared to the old one. The overriding nature of the old visualisation was replaced with an easy to understand timeline, while the general concepts (macro view) can still be read out from the shape of the lines. Micro view is also improved as we can determine contributions of attribute values for any given time.

The detection of concept drift using the stream of explanations did not prove to be suitable for general use based on the initial experiments. It has shown to be hindered by delayed detection times, missed concept drift occurrences, false alarms, high level of parametrization and potential high time complexity. This provides motivation for further experiments in this field, especially because the stream of explanations provides good insight into the model with accordance to the chosen granulation.

The main goal of future research is finding a true adaptation of *IME* explanation methodology to incremental setting, i.e. efficient incremental updates of explanation at the arrival of each new example. Truly incremental explanation methodology would provide us with a stream of explanations of finest granularity. In addition to this result (for each timestamp we get an accurate explanation of the

[5]Considering prior domain knowledge.

current model), a number of new possibilities for visualisation would emerge, particularly those that rely on finely granular data, such as ThemeRiver [6].

References

[1] A. Bifet, G. Holmes, R. Kirkby, B. Pfahringer, and Braun M. Moa: Massive online analysis.

[2] J. Demšar. Explanation of predictive models and individual predictions in incremental learning (in slovene). B.S. Thesis, Univerza v Ljubljani, 2012.

[3] S. Few. *Now You See It: Simple Visualization Techniques for Quantitative Analysis*. Analytics Press, USA, 1st edition, 2009.

[4] J. Gama. *Knowledge Discovery from Data Streams*. Chapman & Hall/CRC, 1st edition, 2010.

[5] D. Haussler. Overview of the probably approximately correct (pac) learning framework, 1995.

[6] S. Havre, B. Hetzler, and L. Nowell. Themeriver: Visualizing theme changes over time. In *Proc. IEEE Symposium on Information Visualization*, pages 115–123, 2000.

[7] L. Nowell, E. Hetzler, and T. Tanasse. Change blindness in information visualization: a case study. In *Information Visualization, 2001. INFOVIS 2001. IEEE Symposium on*, pages 15–22, 2001.

[8] E. S. Page. Continuous inspection schemes. *Biometrika*, 41(1):100–115, 1954.

[9] Gunopulos D. Keogh E. Vlachos M. Das G. Ratanamahatana C.A., Lin J. Mining time series data. In *Data Mining and Knowledge Discovery Handbook*, pages 1049–1077. Springer US, 2010.

[10] R. Sebastião and J. Gama. A study on change detection methods. In *Progress in Artificial Intelligence, 14th Portuguese Conference on Artificial Intelligence, EPIA 2009, Aveiro, Portugal, October 12-15, 2009. Proceedings*, pages 353–264. Springer, 2009.

[11] W. Nick Street and Y. Kim. A streaming ensemble algorithm (sea) for large-scale classification. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '01, pages 377–382, New York, NY, USA, 2001. ACM.

[12] E. Štrumbelj and I. Kononenko. An efficient explanation of individual classifications using game theory. *The Journal of Machine Learning Research*, 11:1–18, 2010.

[13] E. Štrumbelj, I. Kononenko, and M. Robnik Šikonja. Explaining instance classifications with interactions of subsets of feature values. *Data Knowl. Eng.*, 68(10):886–904, October 2009.

Foreground-Background Separation by Feed-forward Neural Networks in Old Manuscripts

Abderrahmane Kefali, Toufik Sari and Halima Bahi
LabGED Laboratory
Computer Science Department, Badji Mokhtar University, BP-12
Annaba, 23000, Algeria
E-mail: {kefali, sari, bahi}@labged.net

Keywords: foreground-background separation, binarization, ANN, MLP, image fusion, PSNR

Received: February 16, 2014

Artificial Neural Networks (ANNs) are widely used techniques in image processing and pattern recognition. Despite of their power in classification tasks, for pattern recognition, they show limited applicability in the earlier stages such as the foreground-background separation (FBS). In this paper a novel FBS technique based on ANN is applied on old documents with a variety of degradations. The idea is to train the ANN on a set of pairs of original images and their respective ideal black and white ones relying on global and local information. We ran several experiments on benchmark and synthetic data and we obtained better results than state-of-the art methods.

Povzetek: V tem prispevku je predlagan nova binarizacija tehnika, ki temelji na umetni nevronski mreži za stare dokumente z različnimi nivoji poslabšanja.

1 Introduction

Important documentary collections exist currently in the libraries, museums and other institutions in pedagogic or sociopolitical matters. Historical documents of old civilizations and public archives are typical examples of such abundance which represent the patrimony, and nation's history. In the last two decades, the scientific community renewed the interest in the restoration of old documents. Several R&D projects were initiated and several research papers and PhD thesis have, consequently, focused on ancient documents processing and analysis. The pattern recognition and image processing researchers are in the core of this interest group.

In the most cases, document image processing should deal with foreground-background separation[1] (FBS). The goal of a FBS algorithm is to extract the relevant information (text, figures, tables, etc.), i.e. "the foreground", from the page, i.e. "the background". Actually, image binarization is critical in the sense that bad separation will cause the loss of pertinent information and/or add useless information. FBS is more difficult for old documents images which have various types of degradations from the digitization process itself, aging effects, humidity, marks, fungus, dirt, etc.

The FBS is generally done using a cut-off value "the threshold" [3][11]. The literature is rich in methods for document image binarization based on thresholding [1][2][3][11]. These methods use different ways to calculate the threshold and are grouped into two main classes: global methods when a single threshold is used for the whole image and local methods when one threshold per image area is computed. Mixtures can be done combining global and local information to find optimal threshold values.

ANNs have contributed since their introduction in the fifties to solve more complex problems in many areas including classification, prediction, approximation, pattern recognition, etc. ANNs are able to generalize, after a learning stage, their behavior on new data they had not "seen" before. In spite of the ANNs' success in various image processing applications, their contribution in image binarization is still limited and have not generated much research [3][4].

In this paper, we exploit the generalization facility of ANNs to devise a novel binarization approach that relies not on thresholding, but on classification using a Multilayer Perceptron (MLP). This assumption can be justified by definition that the binarization is in fact a classification process where the set of image pixels should be divided into two classes "blacks" and "whites" and, for this, ANNs can excel. The benefit of using a supervised neural network for FBS is the reduction of the complexity of the conventional thresholding methods and the availability of large databases for training and validation.

The remainder of this paper is organized as follow. We first give a brief description of kinds of thresholding techniques and focus on ANNs based methods. Next, we describe and detail our proposed approach. After that, we

[1] FBS is also called « binarization » meaning producing black and white (BW) images from grayscale or color ones.

present the evaluation results with several well-known methods, to finish with a conclusion.

2 Related works

Numerous surveys and competitions on document image binarization have been completed not only for comparison purposes but even for categorization, evaluation, and of course for scholars too. According to [3], the binarization techniques can be divided into six groups:

- Histogram-based methods: the methods of this class perform a thresholding based on the form of the histogram. [20][21];
- Clustering-based methods: These methods assign the image pixels to one of the two clusters: object and background. [22][23];
- Entropy-based methods: These algorithms use the information theory to obtain the threshold. [24][25];
- Object attribute-based methods: Find a threshold value based on some similarity measurements between original and binary images. [26][27];
- Spatial binarization methods: find the optimal threshold value taking into account spatial measures. [28];
- Locally adaptive methods: These methods are designed to give a new threshold for every pixel. Several kinds of adaptive methods exist. We find methods based on local gray range [29], local variation [30], etc.

Recently, ANNs have been used for binarization. As stated in [16], most of these works dealt with noisy handwritten document images and non-document images but, however, did not inspect the special case of historical documents.

M. Egmont-Petersen et al., [4] reviewed more than 200 works on image processing with ANNs from which only three were on image binarization. N. Papamarkos et al. in [5] proposed a multithresholding approach using the Principal Component Analysis (PCA) and a Kohonen Self-Organized Feature Map (SOFM) neural network. The input layer of SOFM coded the 255 elements of the gray-level histogram and the output layer is a 1D map in which only one winner neuron is activated for each input vector. In [6], the authors used a feed forward neural network to find the “clean” pixel corresponding to the noisy one as input in conjunction with the median filter value and Rank-Ordered Absolute Differences (ROAD) of the same pixel. We found that the same approach was proposed earlier in [7]. In [31], a PNN network is used but the histogram is compacted. On the other hand, in [32] every pixel is used to obtain the thresholded image.

A combination of a thresholding technique and an artificial neural network for leaf veins extraction is carried out in [12]. Padekas and Papamarkos [13] combined several binarization techniques using a Self-Organizing Map, for the binarization of normal and degraded printed documents for character visualization and recognition.

Yang et al. [14], used text image segmentation method with a neural network for document image processing. In [15], a multi-layer perceptron is used and trained using noisy document images to produce enhanced document images.

In [16], authors combined a global thresholding method, namely the Mass-Difference (MD) thresholding and supervised neural network for selecting a global optimum threshold value. This last is used then to binarize the degraded document image. MD thresholding is first applied to calculate several local optimum threshold values. Afterwards a supervised neural network is trained using these optimum threshold values as its inputs and a single global optimum threshold value as its output. The neural network consists of an input layer having 256 neurons receiving the local threshold values, one hidden layer containing 22 neurons and only one neuron in the output layer which represents the global threshold value.

Làzaro and al. [17] proposed to obtain an optimum threshold for each image using a semantic description of the histogram and a general regression neural network, for high precision OCR algorithms over a limited number of document types. The histogram of the input image is first smoothed in order to eliminate false minima and maxima and its discrete derivate is found. Using a polygonal version of the derivate and the smoothed histogram, a new description of the histogram is calculated. Once this last is generated, a semantic description is inferred. The obtained semantic description is used by a general regression neural network to obtain the optimal threshold value.

In [18], authors used multi-layer perceptron (MLP) as a threshold transfer to select the visually satisfied threshold by a modified back propagation algorithm. The proposed technique starts with the histogram of the contrasted image. The histogram is scaled to [0, 1] and divided into 128. The three-layer MLP has an input layer of 128 neurons, two hidden layers of 256 and 512 neurons respectively and an output layer with 128 neurons.

[19] Introduced three neural based binarization techniques. The three approaches start with a Self Organizing Map (SOM) trained over a part of the image in order to extract its most representative grey levels or colors. The SOM inputs are the pixels of the image. Then, the classification is done differently in the three approaches. In the first approach, the neurons of the SOM are segmented into N regions (for N classes) using the Kmeans algorithm [9]. Once the pixels are clustered, the SOM is used to classify the pixels of the entire image. Each pixel is given as input to the SOM in order to be classified. The second approach is applied on color images. In this approach an MLP is used where the inputs are the SOM neurons, and its outputs are the classes of the same neurons. After training, the MLP is applied on the whole image to classify each pixel. In the third approach, Sauvola or Niblack thresholds are applied on the neurons of the trained SOM, and a global threshold is extracted. This global threshold is applied on the entire image.

Although the binarization record is long and will continue to lengthen (see ICDAR 2013 competitions),

few comparison studies were found. A large discussion on performances of the works mentioned above is not possible at this stage. A good comparison should be done, in our opinion, but on standard and benchmark databases of document images and using well renowned performance metrics.

3 Proposed method

In this section we will describe a new method for binarizing images of old documents. The proposed method may be placed among the hybrid class. In this method, the separation of image pixels to "blacks" or "whites" is performed by a Multilayer Perceptron (MLP) trained with back-propagation (see [49] for more details). In this approach, the MLP does not compute or learn any threshold but runs a direct binarization by classifying the image pixels into two classes. As we described above the binarization is a kind of two-class classification problem.

The MLP is an ANN with supervised learning dedicated especially for classification purposes. It has the ability of separating non-linearly separable classes of patterns. We have chosen to use MLPs as gray-levels of pixels may overlap in some cases. Even if this is not always true, MLPs go surely further than many other classifiers [19]. The motivation of using learning-based approach for FBS is that, as detailed in Figure 1, the pages of one same volume have similar defects as they are written in the same type of paper, in the same period of time and are conserved in the same conditions. As a result, one binarization algorithm will equally succeed on all the pages of the same manuscript. Therefore, we can train the algorithm on a limited number of pages, or some specific areas from different pages, and it can perform well on the other pages. For other volumes, we can re-train the classifier to adapt to the new data. The same scheme is also possible for one single page; we can train the algorithm on only the data from some areas of the page and it generalizes its behavior on the remainder of the page.

(a) *(b)* *(c)*

Figure 1: (a) a collection of documents in the same conservation conditions. (b) & (c) different pages of the same volume with similar degradations.

For each pixel, global and local information from the neighborhood are used to train the MLP. Local information is the grey values of the pixel *p* with those of its neighbors from an $N \times N$ window centered on *p*. In addition, we introduce global information, namely the global mean (*M*) and global standard deviation (*S*) of the whole image. Indeed, as noted in [18], these two last and other statistical parameters of the image are widely used by the most binarization methods, [22] [23] for instance, to compute the thresholds.

The MLP should then output 0 for black or 1 for white. See Figure 2.

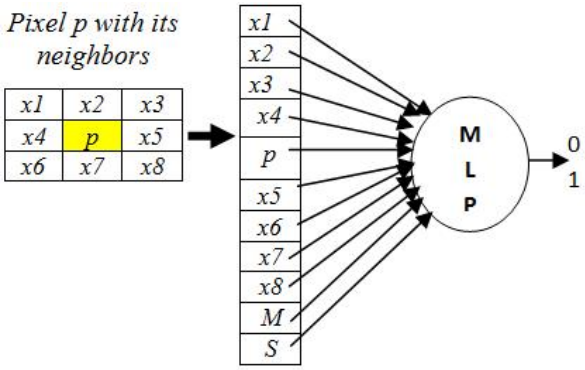


Figure 2: MLP for classifying a pixel p according to its current value and of those of its neighbors in a 3×3 window.

The proposed approach consists of four steps: *image preprocessing*, *MLP definition*, *MLP training*, and finally *Binarization* step. The block diagram of the approach is shown in Figure 3.

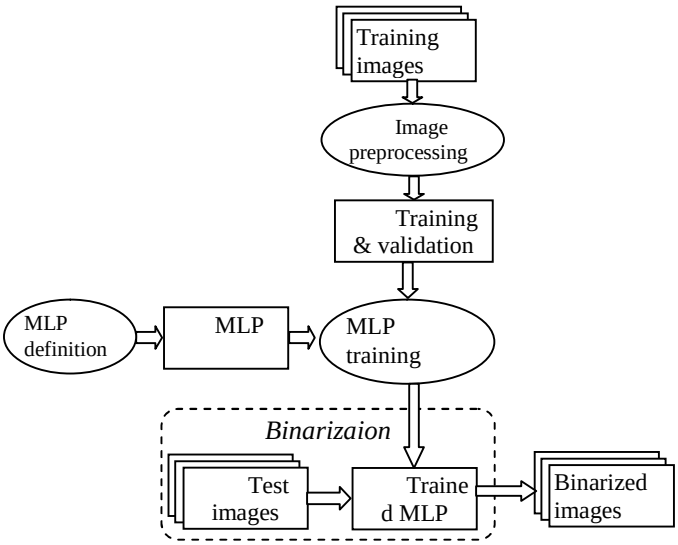


Figure 3: Block diagram of the proposed method.

3.1 MLP definition

The first thing we need to do is to set the “adequate” structure of the neural network (number of hidden layers and number of neurons per layer).

As we said, we preferred to work with Multilayer Perceptron which showed proven abilities in classification problems. Despite the importance of the optimal topology of an MLP for a given problem, it is not always easy to devise and almost not necessary.

Kolmogorov [50] showed that all continuous functions of *n* variables have an exact representation in terms of finite superpositions and compositions of a small number of functions of one variable [8]. In terms of networks this means that every continuous function of

many variables can be a network with two hidden layers [52]. In addition, according to G.V. Cybenco in [8] *"a MLP network that has only one hidden layer is able to approximate almost any type of non linear mapping"*.

So, we used the trial-and-error procedure to find the ideal topology of our MLP. We devised an MLP with a single output layer corresponding to the binary value of the pixel (0 or 1). The number of inputs is the number of pixels in the window (9 for a 3×3 window, 25 for 5×5, etc.), in addition to the Mean (*M*) and the standard deviation (*S*) of the whole image. The activation function chosen is the sigmoid *function* defined by:

$$f(x) = \frac{1}{1+e^{-ax}} \ . \qquad (1)$$

However, one question remains: which window size will give better results? To answer this question, we tested several window sizes, with different neural networks of varying number of layers, and neurons in each layer; and trained all these configurations and evaluate for each configuration its performance on the test set, using several evaluation measures. The obtained results show that the optimal configuration is:

- Window size of 3×3 and therefore the number of inputs is 11,
- One hidden layer of 11 neurons,
- One neuron as output.

3.2 Image preprocessing

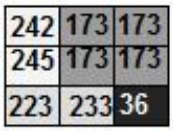
This phase has as goal the preparation and the extraction of training (and validation) data from a sample of training images and it runs offline.

3.2.1. Training and validation data

To guarantee a high accuracy, the MLP should be trained on a huge set of patterns in order to determine with high precision its free parameters (the links weights).

In practice, when using only a training set produces, in the most cases, the phenomenon of *over-learning*; where the ANN learns the training data but is not able to generalize to other data not seen during the training phase. To overcome this problem, we use another set of data called 'validation set'. During the training phase, at every epoch we check in addition to the learning error the validation error and compare it to previous values to determine the moment of performance drop.

The training and validation sets are composed of several input vectors with the corresponding desired output. The size of the input vector is the number of input neurons of the MLP. In our case, each vector represents the gray values of a pixel with that of those neighborhoods, in addition to the global mean (*M*) and global standard deviation (*S*) of the whole image (Figure 4). The corresponding expected output is the binary value of the related pixel in the ground truth image (the black and white image). As the pixel values are in [0, 255], the data should be normalized or scaled to the interval [0, 1] (0 for black and 1 for white) to ensure the proper functioning of the neuron in sensitive regions of the activation function.



<242,173,173,245,**173**,173,223,233,36, 185.66, 64.58>

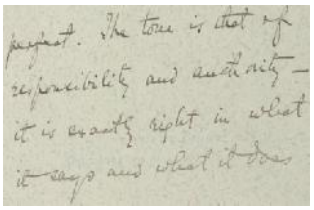
Figure 4: Input vector for one pixel in a 3×3 window.

3.2.2. Images of training and validation

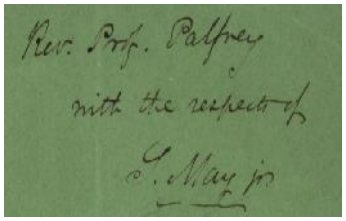
We used two sets of images for creating the training and validation data. The first is a public set composed of document images from the four collections proposed within the context of the competitions DIBCO 2009[2], H-DIBCO 2010[3], DIBCO 2011[4] and H-DIBCO 2012[5].

These four collections contain a total of 50 real documents images (37 handwritten and 13 printed) coming from the collections of several libraries, with the associated ground truth images. All the images contain representative degradations which appear frequently (e.g. variable background intensity, shadows, smear, smudge, low contrast, bleed-through). Figure 5 show some images from these collections.

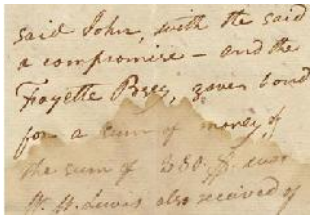
(a)



(b)



(c)



(d)

Figure 5: Images taken from the collections of: (a) DIBCO 2009, (b) H-DIBCO 2010, (c) DIBCO 2011, (d) H-DIBCO 2012.

The second set of images is a synthetic collection prepared in order to include most of the degradations in old documents. It is created by applying the *image mosaicing by superimposing technique for blending* [33]. The idea is as follow, we start with some images of documents in black & white, which represent the ground truth, and with some backgrounds extracted from old documents and we apply a *fusion procedure* to get as many different images of old documents. However, P. Stathis et *al.*, in [10] proposed two different techniques for the blending: *maximum intensity* and *image averaging*. We adopt to use the *image averaging technique* in order to have a more natural result.

[2] http://users.iit.demokritos.gr/~bgat/DIBCO2009/benchmark/
[3] http://www.iit.demokritos.gr/~bgat/H-DIBCO2010/benchmark
[4] http://utopia.duth.gr/~ipratika/DIBCO2011/benchmark
[5] http://utopia.duth.gr/~ipratika/HDIBCO2012/benchmark

The Mosaicing process used can be summarized by the following pseudo-code:

```
Input:  GT: the ground truth image
        BG: the background
Output: R:  the resulting image
For Each pixel (i,j)
   If BG(i,j) is More_Darker_Than GT(i,j) Then
     R(i,j) = BG (i,j)
   Else R(i,j) = (GT(i,j)+BG(i,j))/2
End For
```

We note here that we used a large amount of backgrounds with different varieties of degradations (transparency effects, holes, stains, etc.) to allow the MLP to learn from a vast diversity of possible cases (Figure 6).

(a) *(b)* *(c)*

Figure 6: Images of documents obtained by fusing binary images and backgrounds. (a) background from old documents, (b) ground truth binary image, (c) the resulting synthetic images.

3.3 MLP training

In this phase, the MLP is trained using the training and validation data prepared before. For that, we used back propagation algorithm. It is a supervised learning algorithm, where the system is provided with samples of inputs and the corresponding expected, or desired, output values.

The training process is done as following: we introduce the training vectors to the MLP, and we calculate the error between the outputs of the MLP (estimated output) and the real data (desired outputs). Next, we update the weights of all neurons. After that, we provide the validation vectors to the MLP, and we calculate the validation error (as before), without updating the neurons weights. The process is repeated in order to minimize both the learning and the validation error. The training is interrupted when the validation error begins to increase.

After training the neural network learns (usually) to provide the correct output value when it is presented with the input value only.

3.4 Binarization

Once learning is achieved, the last phase of the proposed method is the use of the trained MLP to binarize different document images. It is done by running the trained neural network with one forward pass using the final training weights. For each pixel of the processed image, we provide the MLP the feature vector as in the training phase and it should output a binary value for the pixel.

4 Experimentation and results

Our application was developed in Java with the framework *Neuroph*[6].

As we said before, we used two sets of documents for the training and validation. The first set is a public collection of a total of 50 real documents images (37 handwritten and 13 printed) coming from the collections of several libraries, with the associated ground truth. The second set is composed of 240 synthetic images of documents constructed by our fusion algorithm from 20 different backgrounds and 12 binary images.

As a first attempt, we used 15 images of the first set and 70 images of the second set for training, 10 images of the first set and 30 images of the second set for validation, and the remainders for testing. We also used all the pixels in all images of training and validation sets. This was not practicable because the images are larger (average 783,000 pixels) resulting in a massive training set of about 66,555,000 vectors. Then, we considered selecting a portion of vectors to use in training and validation. Thus, we selected randomly 500 vectors for training and validation image, resulting 42,500 vectors for training and 20,000 others vectors for validation.

During the training we calculate at each time the training and validation errors. As common, the training error decreases continuously and gradually but the validation error decreases at the beginning and then starts to deteriorate which mean that over-learning had occurred and so we should stop learning process. For our case, this happened at the epoch #25,382 (Figure 7).

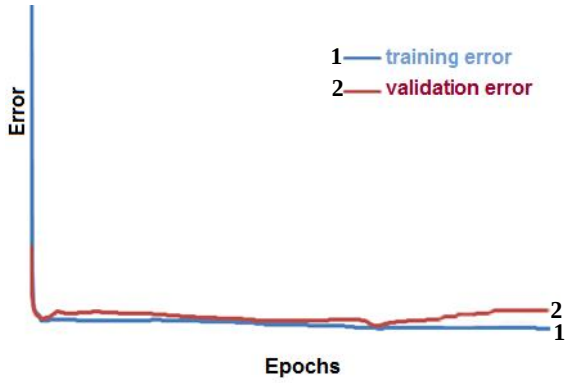


Figure 6: Errors of training and validation.

To assess the generalization ability of our neural network, we tested it on the testing collection containing 165 images of new old documents that were not presented to the MLP (25 of the first set and 140 of the second set), and compared the resulting binary images with the corresponding ground truth ones. The comparison is done quantitatively by using standard measures that have been widely used for evaluation purposes, especially in DIBCO 2009 [2], H-DIBCO 2010 [1], DIBCO 2011 [53], H-DIBCO 2012 [54] competitions. These measures consist of: FMeasure, PSNR, NRM, MPM, and DRD.

[6] http://neuroph.sourceforge.net

Noting *TP*, *TN*, *FP*, *FN* the True positive, True Negative, False positive and False negative values, respectively.

i) FMeasure

F-Measure was introduced first by Chinchor in [55].

$$FM = \frac{2 \times \mathrm{Re}call \times \mathrm{Pr}ecision}{\mathrm{Re}call + \mathrm{Pr}ecision}. \quad (2)$$

Where:

$$\mathrm{Re}call = \frac{TP}{TP + FN} \quad \text{and} \quad \mathrm{Pr}ecision = \frac{TP}{TP + FP}$$

ii) PSNR (Peak Signal to Noise Ratio)

PSNR is a similarity measure between two images. However, the higher the value of PSNR, the higher the similarity of the two images [2][34].

$$PSNR = 10.\log\left(\frac{C^2}{MSE}\right). \quad (3)$$

Where: $MSE = \frac{\sum_{x=1}^{M}\sum_{y=1}^{N}\left(I(x,y) - I_2(x,y)\right)^2}{MN}$

I and I_2 represents the two images matched. M and N there height and width respectively. C the difference between foreground and background (here 255).

iii) NRM (Negative Rate Metric)

NRM is based on the pixel-wise mismatches between the Ground Truth and the binarized image [51]. It combines the false negative rate NR_{FN} and the false positive rate NR_{FP}. It is denoted as follows:

$$NRM = \frac{NR_{FN} + NR_{FP}}{2}. \quad (4)$$

With : $NR_{FN} = \frac{FN}{FN + TP}$ and $NR_{FP} = \frac{FP}{FP + TN}$

The better binarization quality is obtained for lower NRM.

iv) MPM (Misclassification Penalty Metric)

The Misclassification penalty metric MPM evaluates the binarization result against the Ground Truth on an object-by-object basis [51].

$$MPM = \frac{MP_{FN} + MP_{FP}}{2}. \quad (5)$$

where : $MP_{FN} = \frac{\sum_{i=1}^{FN} d_{FN}^{i}}{D}$ and $MP_{FP} = \frac{\sum_{j=1}^{FP} d_{FP}^{j}}{D}$

d_{FN}^{i} and d_{FP}^{j} denote the distance of the i^{th} false negative and the j^{th} false positive pixel from the contour of the Ground Truth segmentation. The normalization factor D is the sum over all the pixel-to-contour distances of the Ground Truth object. A low MPM score denotes that the algorithm is good at identifying an object's boundary.

v) DRD (Distance Reciprocal Distortion Metric)

DRD is an objective distortion measure for binary document images, and it was proposed by Lu et al. in [34]. This measure properly correlates with the human visual perception and it measures the distortion for all the S flipped pixels as follows:

$$DRD = \frac{\sum_{k=1}^{S} DRD_k}{NUBN}. \quad (6)$$

NUBN is the number of the non-uniform 8×8 blocks in the GT image.

DRD_k is the distortion of the k^{th} flipped pixel of coordinate *(x, y)* and it is calculated using a 5×5 normalized weight matrix W_{Nm}. This last is defined in [34] as follow:

$$W_{Nm}(i,j) = \frac{W_m(i,j)}{\sum_{i=1}^{m}\sum_{j=1}^{m} W_m(i,j)}$$

Such as:

$$W_m(i,j) = \begin{cases} 0, & \text{for } i = i_C \text{ and } j = j_C \\ \dfrac{1}{\sqrt{(i-i_C)^2 + (j-j_C)^2}}, & \text{otherwise.} \end{cases}$$

With $m = 5$, and $i_C = j_C = (1 + m) / 2$.

DRD_k is given as follow (eq.7):

$$DRD_k = \sum_{i=-2}^{2}\sum_{j=-2}^{2} \left| I_{GT}(x+i, y+j) - I_B(x,y) \right| \times W_{Nm}(i+2, j+2)$$

We also compared our method with several well known state-of-the-art methods of document binarization [3][10][11]. For the local methods, we reimplemented, all used values of parameters are those indicated in the original papers. The average results obtained with all compared methods over each test set are summarized in Table.1. The average results between the two sets are shown in Table.2. The final ranking of the compared methods is shown in Table.3, which also summarizes the partial ranks of each method according to each evaluation measure and the sum of ranks. Afterwards, We provide in Fig.8 graphs representing the average performance of the compared methods in terms of FMeasure and PSNR.

From the Tables and Fig.8, our proposed technique is ranked second after Sauvola and Pietikainen method for both test sets, and it has good performances in terms of FMeasure and PSNR. This result is due to the generalization ability of the MLP despite of the characteristic of degradation and the variation of noise types present in the documents.

	First test set					Second test set				
Method	**FM**	**PSNR**	**NRM**	**MPM**	**DRD**	**FM**	**PSNR**	**NRM**	**MPM**	**DRD**
Otsu [22]	0.8571	15.886	0.0628	0.1915	1.755	0.6245	15.101	0.1590	2.8783	4.6768
ISODATA [37]	0.8544	15.734	0.0632	0.1938	1.833	0.6153	15.059	0.1666	2.9310	4.7320
Kittler and Illingworth [23]	0.4673	6.2878	0.1630	9.7698	26.13	0.4239	8.3621	0.2163	17.727	28.682
Kapur and al.[24]	0.8432	14.989	0.0509	0.2673	2.161	0.3618	11.493	0.3096	3.3474	7.6726
Mello and Lins [35]	0.6496	12.708	0.2055	0.3465	3.051	0.3574	13.713	0.3314	0.7512	4.3837
Tsallis Entropy Based Algorithms [36]	0.5203	13.092	0.2712	0.0901	3.092	0.1387	11.532	0.4547	0.3561	3.9472
Iterative global thresholding (IGT) [38]	0.7622	13.683	0.1162	1.7347	4.215	0.6049	13.637	0.1711	2.3889	4.0455
Niblack [30]	0.5321	7.7542	0.1259	8.3005	17.85	0.3996	7.5398	0.1937	9.4157	17.059
Sauvola and Pietikainen. [39]	0.8360	15.537	0.0938	0.2486	1.634	0.6718	16.130	0.1835	0.7630	2.2288
Nick [40]	0.7764	13.882	0.0977	1.5542	4.116	0.6778	15.477	0.1656	0.9653	2.5658
Feng and Tan. [41]	0.7798	13.826	0.0820	2.2608	5.016	0.6744	15.148	0.1576	1.0867	2.7634
Bernsen [29]	0.5150	8.1056	0.1807	10.936	19.62	0.4081	8.1610	0.2069	8.7728	14.517
Meen-Gradient (Leedham and al.) [42]	0.7914	13.872	0.1009	0.2631	2.834	0.6023	12.832	0.1858	1.2630	3.7058
Cavalcanti and al. [43]	0.4926	11.928	0.3117	0.0807	4.255	0.3443	12.815	0.3454	0.8949	4.0059
Tabatabei and Bohlool. [44]	0.8002	14.744	0.0671	2.4013	5.309	0.6377	15.660	0.1572	2.2360	3.6973
Gangamma and al.[45]	0.6714	12.879	0.1765	4.1804	6.549	0.6037	14.317	0.2179	1.0963	2.4942
Improved IGT [46]	0.7530	13.695	0.1364	1.3784	3.751	0.5980	13.486	0.1921	1.8568	3.7286
Using Local Max and Min [47]	0.7891	14.342	0.0971	2.2913	5.439	0.5713	11.622	0.1835	1.3603	4.5757
Sari and al.[48]	0.8362	15.628	0.0719	0.7248	1.970	0.4979	9.4425	0.1764	4.3272	10.974
Proposed method	0.8436	15.555	0.0599	0.7468	2.597	0.6608	15.396	0.1456	1.806	2.9860

Table 1: Average results from different binarization methods on each test set.

Method	**FM**	**PSNR**	**NRM**	**MPM**	**DRD**
Otsu [22]	0.7408	15.4939	0.1109	1.5349	3.2159
ISODATA [37]	0.7348	15.3966	0.1149	1.5624	3.2825
Kittler and Illingworth [23]	0.4456	7.3250	0.1897	13.7486	27.406
Kapur and al.[24]	0.6025	13.2417	0.1803	1.8073	4.9168
Mello and Lins [35]	0.5035	13.2112	0.2684	0.5488	3.71735
Tsallis Entropy Based Algorithms [36]	0.3295	12.3124	0.3629	0.2231	3.5196
Iterative global thresholding (IGT) [38]	0.6835	13.6609	0.1437	2.0618	4.13025
Niblack [30]	0.4658	7.6470	0.1598	8.8581	17.4545
Sauvola and Pietikainen. [39]	0.7539	15.8343	0.1386	0.5058	1.9314
Nick [40]	0.7271	14.6801	0.1316	1.2597	3.3409
Feng and Tan. [41]	0.7271	14.4877	0.1198	1.6737	3.8897
Bernsen [29]	0.4615	8.1333	0.1938	9.8547	17.0685
Meen-Gradient (Leedham and al.) [42]	0.6969	13.3522	0.1433	0.7630	3.2699
Cavalcanti and al. [43]	0.4185	12.3722	0.3286	0.4878	4.13045
Tabatabei and Bohlool. [44]	0.7190	15.2025	0.1122	2.3186	4.50315
Gangamma and al.[45]	0.6376	13.5984	0.1972	2.6384	4.5216
Improved IGT [46]	0.6755	13.5911	0.1643	1.6176	3.7398
Using Local Max and Min [47]	0.6802	12.9824	0.1403	1.8258	5.00735
Sari and al.[48]	0.6670	12.5353	0.1241	2.5260	6.472
Proposed method	0.7522	15.4759	0.1028	1.2764	2.7915

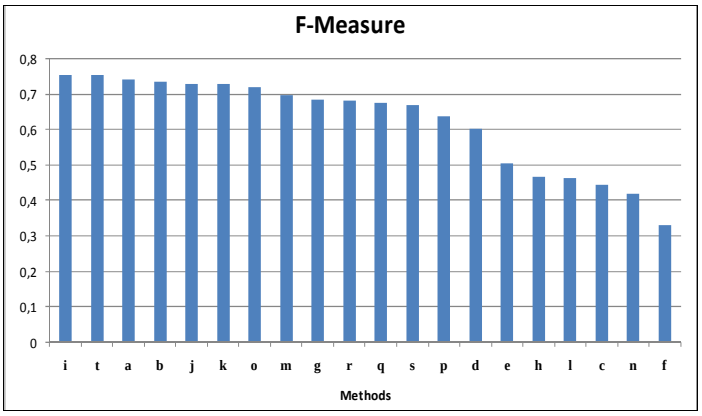
Table 2: Average results from different binarization methods between the two test sets.

5 Conclusion

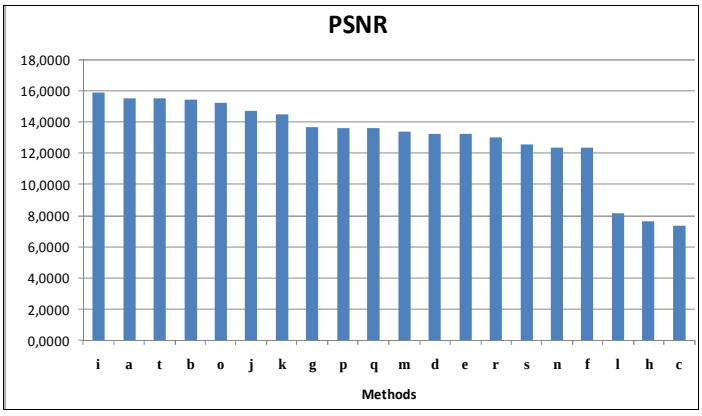
Document binarization is an important and crucial step in all systems of document analysis and recognition. This task becomes more difficult for historical documents containing various types of noises and degradations. In this paper, we proposed a new ANN based method for old document binarization. The purpose of using ANN, and especially Multilayer Perceptrons, for image binarization is to fill the lack of employing the techniques of soft computing and machine learning in such problem, and to take advantages of the generalization abilities of the MLP. Indeed, as confirmed by the experimentation results, the MLP presented a reliable behavior for the complex task of foreground/background separation from significantly degraded document images. Many extensions are possible and we will continue to enhance the proposed method. Other experimentations are needed in order to identify the applicability of the proposed solution in mobile devises of real life utilization.

Rank	Method	FM	PSNR	NRM	MPM	DRD	Sum of ranks
1	Sauvola and Pietikainen. [39]	1	1	8	3	1	14
2	Proposed method	2	3	1	7	2	15
3	Otsu [22]	3	2	2	8	3	18
4	ISODATA [37]	4	4	4	9	5	26
5	Nick [40]	5	6	7	6	6	30
6	Meen-Gradient (Leedham and al.) [42]	8	11	10	5	4	38
7	Feng and Tan. [41]	6	7	5	11	10	39
8	Tabatabei and Bohlool. [44]	7	5	4	15	13	44
9	Improved IGT [46]	11	10	13	10	9	53
10	Iterative global thresholding (IGT) [38]	9	8	11	14	12	54
11	Mello and Lins [35]	15	13	18	4	8	58
12	Using Local Max and Min [47]	10	14	9	13	16	62
13	Tsallis Entropy Based Algorithms [36]	20	17	20	1	7	65
14	Sari and al.[48]	12	15	6	16	17	66
15	Kapur and al.[24]	14	12	14	12	15	67
16	Cavalcanti and al. [43]	19	16	19	2	11	67
17	Gangamma and al.[45]	13	9	17	17	14	70
18	Niblack [30]	16	19	12	18	19	84
19	Bernsen [29]	17	18	16	19	18	88
20	Kittler and Illingworth [23]	18	20	15	20	20	93

Table 3: Final ranking of the compared methods on the two test sets.



(a)



(b)

A	Otsu
b	ISODATA
c	Kittler and Illingworth
d	Kapur and al.
e	Mello and Lins
f	Tsallis Entropy Based Algorithms
g	Iterative global thresholding (IGT)
h	Niblack
i	Sauvola and Pietikainen.
j	Nick
k	Feng and Tan.
l	Bernsen
m	Meen-Gradient (Leedham and al.)
n	Cavalcanti and al.
o	Tabatabei and Bohlool.
p	Gangamma and al.
q	Improved IGT
r	Using Local Max and Min
s	Sari and al.
t	Proposed method

Figure 7: Graphs showing the performance of the compared binarization methods in terms of (a) FMeasure and (b) PSNR.

References

[1] I. Pratikakis, B. Gatos, K. Ntirogiannis (2010). H-DIBCO 2010 - Handwritten Document Image Binarization Competition. *In proceedings of the International Conference on Frontiers in Handwriting Recognition ICFHR*, pp. 727-732.

[2] B. Gatos, K. Ntirogiannis, I. Pratikakis (2009). DIBCO 2009: document image binarization contest.

International Journal of Document Analysis and Recognition IJDAR, Vol.14, No. 1, pp. 35-44.

[3] M. Sezgin, B. Sankur (2004). Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, Vol. 13, No. 1, pp. 146-165.

[4] M. Egmont-Petersen, D. de Ridder, H. Handels (2002). Image processing with neural networks - a review. *Pattern Recognition*, Vol. 35, No. 10, pp. 2279-2301.

[5] N. Papamarkos, C. Strouthopoulos, I. Andreadis (2000). Multithresholding of color and gray-level images through a neural network technique. *Image and Vision Computing*, Vol. 18, No. 3, pp. 213-222.

[6] G. Kaliraj, S. Baskar (2010). An efficient approach for the removal of impulse noise from the corrupted image using neural network based impulse detector. *Image and Vision Computing*, Vol. 28, No. 3, pp. 458-466.

[7] H. Kong, L. Guan (1998). A noise-exclusive adaptive filtering framework for removing impulse noise in digital images. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, Vol. 45, No. 3, pp. 422-428.

[8] G. Cybenco (1989). Approximation by superposition of a sigmoidal function. *Mathematics of control, Signals, and Systems*, Vol. 2, No. 4, pp. 303-314.

[9] J. B. MacQueen (1967). Some methods for classification and analysis of multivariate observations. *In Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297.

[10] P. Stathis, E. Kavallieratou, N. Papamarkos (2008). An Evaluation Survey of Binarization Algorithms on Historical Documents. *In proceedings of the 19th International Conference on Pattern Recognition ICPR*, pp. 742-745.

[11] A. Kefali, T. Sari, M. Sellami (2010). Evaluation of several binarization techniques for old Arabic documents images. *In proceedings of the First International Symposium on Modeling and Implementing Complex Systems MISC.* Constantine, Algeria, pp. 88-99.

[12] H. Fu, Z. Chi (2006). Combined Thresholding and Neural Network Approach for Vein Pattern Extraction from Leaf Images. *IEE Proceedings-Vision, Image and Signal Processing*, Vol. 153, No. 6, pp. 881-892.

[13] E. Badekas, N. Papamarkos (2007). Optimal Combination of Document Binarization Techniques Using a Self Organizing Map Neural Network. *Engineering Applications of Artificial Intelligence*, Vol. 20, No. 1, pp. 11-24.

[14] Y. Yang, K. Summers, M. Turner (2004). A Text Image Enhancement System Based on Segmentation and Classification Method. *In Proceedings of the First ACM Workshop on Hardcopy Document Processing HDP*, pp. 33-40.

[15] J.L. Hidalgo, S. Espana, M.J. Castro. J.A. Perez (2005). Enhancement and Cleaning of Handwritten Data by Using Neural Networks. *Lecture Notes in Computer Science*, Vol.3522. Springer-Verlag, Berlin Heidelberg New York, pp. 376-383.

[16] A. Khashman, B. Sekeroglu (2007). Global Binarization of Document Images Using a Neural Network. *Third International IEEE Conference on Signal-Image Technologies and Internet-Based System SITIS*, Shanghai, pp. 665-672.

[17] J. Lazaro, J.L Martin, J. Arias, A. Astarloa, C. Cuadrado (2010). Neuro Semantic Thresholding for High Precision OCR Applications. *Image and Vision Computing*, Vol. 28, No. 4, pp. 571-578.

[18] T. Chen, M. Takagi (1993). Image Binarization by Back Propagation Algorithm. *International Archives of Photogrammetry and Remote Sensing*, Vol.29, pp. 345-345.

[19] H. Hamza, E. Smigiel, A. Belaid (2005). Neural based binarization techniques. *In Proceedings of the 8th International Conference on Document Analysis and Recognition ICDAR*, pp. 317-321.

[20] M.I. Sezan (1990). A peak detection algorithm and its application to histogram-based image data reduction. *Computer Vision, Graphics, and Image Processing*, Vol. 49, No. 1, pp. 36-51.

[21] A. Rosenfeld, P. de la Torre (1983). Histogram concavity analysis as an aid in threshold selection. *IEEE Transactions on System, Man, and Cybernetics*, Vol. 13, No. 2, pp. 231-235.

[22] N. Otsu (1979). A thresholding selection method from grayscale histogram. *IEEE Transactions on System, Man, and Cybernetics*, Vol. 9, pp. 62-66.

[23] J. Kittler, J. Illingworth (1986). Minimum error thresholding. *Pattern Recognition*, Vol. 19, No. 1, pp. 41-47.

[24] J.N. Kapur, P.K. Sahoo, AK.C. Wong (1985). A new method for gray-level picture thresholding using the entropy of the histogram. *Computer Vision, Graphics, and Image Processing*, Vol. 29, No. 3, pp. 273-285.

[25] H.D. Cheng, J.R. Chen, J. Li (1998). Threshold selection based on fuzzy c-partition entropy approach. *Pattern Recognition*, Vol. 31, No. 7, pp. 857-870.

[26] L. Hertz, R.W. Schafer (1988). Multilevel thresholding using edge matching. *Computer Vision, Graphics, and Image Processing*, Vol. 44, No. 3, pp. 279-295.

[27] L.K. Huang, M.J.J. Wang (1995). Image thresholding by minimizing the measures of fuzziness. *Pattern Recognition*, Vol. 28, No. 1, pp. 41-51.

[28] A.S. Abutableb (1989). Automatic thresholding of gray-level pictures using two-dimensional entropy. *Computer Vision, Graphics, and Image Processing*, Vol. 47, No. 1, pp. 22-32.

[29] J. Bernsen (1986). Dynamic thresholding for gray-level images. *In Proceedings of the 8th International Conference on Pattern Recognition ICPR*, Paris, French, pp. 1251-1255.

[30] W. Niblack (1985). *An introduction to Digital Image Processing*. Strandberg Publishing Company, Birkeroed, Denmark.

[31] K.L. Chung, W.Y. Chen (2003). Fast Adaptive PNN-Based Thresholding Algorithms. *Pattern Recognition*, Vol. 36, No. 12, pp. 2793-2804.

[32] C.Y. Chang, P.C. Chung (2001). Medical Image Segmentation Using a Contextual-Constraint-Based Hopfield Neural Cube. *Image and Vision Computing*, Vol.19, No. 9-10, pp. 669-678.

[33] L.G. Brown (1992). A survey of Image Registration Techniques. *ACM Computing Surveys*, Vol. 24, No. 4, pp. 325-376.

[34] H. Lu, A.C. Kot, Y.Q. Shi (2004). Distance-Reciprocal Distortion Measure for Binary Document Images. *IEEE Signal Processing Letters*, Vol. 11, No. 2, pp. 228-231.

[35] C.A.B. Mello, R.D. Lins (2000). Image segmentation of historical documents. *Visual 2000*, Mexico City, Mexico, Vol. 30.

[36] C.A.B. Mello, A.L.I.D. Oliveira, A. Sanchez (2008). Historical Document Image Binarization. *In proceedings of the International Conference on Computer Vision Theory and Applications*, Funchal, Portugal, Vol. 1, pp. 108-113.

[37] F.R.D. Velasco (1980). Thresholding using the isodata clustering algorithm. *IEEE Transaction on Systems, Man and Cybernetics*, Vol. 10, No. 11, pp. 771-774.

[38] E. Kavallieratou (2005). A Binarization Algorithm Specialized on Document images and Photos. *In Proceedings of the 8th International Conference on Document Analysis and Recognition ICDAR*, pp. 463-467.

[39] J. Sauvola, M. Pietikainen (2000). Adaptive document image binarization. *Pattern Recognition*, Vol. 33, No. 2, pp. 225-236.

[40] K. Khurshid, I. Siddiqi, C. Faure, N. Vincent (2009). Comparison of Niblack inspired Binarization methods for ancient documents. *In Proceedings of the 16th Document Recognition and Retrieval DRR*, USA.

[41] M.L. Feng, Y.P. Tan (2004). Adaptive binarization method for document image analysis. *In Proceedings of the IEEE International Conference on Multimedia and Expo ICME*, pp. 339-342.

[42] G. Leedham, C. Yan, K. Takru, J.H.N. Tan, L. Mian (2003). Comparison of Some Thresholding Algorithms for Text/Background Segmentation in Difficult Document Images. *In Proceedings of the 7th International Conference on Document Analysis and Recognition ICDAR*, Vol.2, pp. 859-864.

[43] G.D.C Cavalcanti, E.F.A. Silva, C. Zanchettin, B.L.D. Bezerra, R.C. Doria, J.C.B. Rabelo (2006). A Heuristic Binarization Algorithm for Documents with Complex Background. *In Proceedings of the International Conference on Image Processing ICIP*, pp. 389-392.

[44] S.A. Tabatabaei, M. Bohlool (2010). Novel method for binarization of badly illuminated document images. *In Proceedings of the IEEE 17th International Conference on Image Processing ICIP*, Hong Kong, pp. 3573-3576.

[45] B. Gangamma, M.K. Srikanta (2011). Enhancement of Degraded Historical Kannada Documents. *International Journal of Computer Applications*, Vol.29, No.11, pp. 1-6.

[46] E. Kavallieratou, S. Stathis (2006). Adaptive Binarization of Historical Document Images. *In Proceedings of the 18th Conference on pattern recognition ICPR*, pp. 742-745.

[47] B. Su, S. Lu, C.L. Tan (2010). Binarization of Historical Document Images Using the Local Maximum and Minimum. *In Proceedings of the 9th IAPR International Workshop on Document Analysis Systems DAS*, Boston, MA, USA, pp. 159-166.

[48] T. Sari, A. Kefali, H. Bahi (2012). An MLP for binarizing images of old manuscripts. *In Proceedings of the International Conference on Frontiers in Handwriting Recognition ICFHR*, pp. 247-251.

[49] S. B. Kotsiantis (2007). Supervised Machine Learning: A Review of Classification Techniques. *Informatica*, Vol.31, pp. 249-268.

[50] A.N. Kalmogorov (1957). On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition. *Dokl. Acad. Nauk SSR*, 114, pp. 953-956

[51] J. Aguilera, H. Wildenauer, M. Kampel, M. Borg, D. Thirde, J. Ferryman (2005). Evaluation of motion segmentation quality for aircraft activity surveillance. *In Proceedings of the Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance VS-PETS*, pp. 317-324.

[52] F. Girosi, T. Poggio (1989). Representation properties of networks: Kolmogorov's theorem is irrelevant. *Neural Computation*, Vol 2, No. 4, pp. 465-469.

[53] I. Pratikakis, B Gatos, K. Ntirogiannis (2011). ICDAR 2011 Document Image Binarization Contest (DIBCO 2011). *In Proceedings of the 2011 International Conference on Document Analysis and Recognition ICDAR*, Beijing, China, pp. 1506-1510.

[54] I. Pratikakis, B. Gatos, K. Ntirogiannis (2012). ICFHR 2012 competition on handwritten document image binarization (H-DIBCO 2012), *In Proceedings of the International Conference on Frontiers in Handwriting Recognition ICFHR*, pp. 817-822.

[55] N. Chinchor (1992). MUC-4 Evaluation Metrics. *In Proceedings of the Fourth Message Understanding Conference*, pp. 22–29.

An Online Compression Algorithm for Positioning Data Acquisition

Wei Pan, Chunlong Yao, Xu Li and Lan Shen
School of Information Science and Engineering, Dalian Polytechnic University, Dalian 116034, China
E-mail: yaocl@dlpu.edu.cn

Keywords: location data acquisition systems, positioning data, trajectory compression, trajectory recovery

Received: September 17, 2014

Positioning data are usually acquired periodically and uploaded to the server via wireless network in the location data acquisition systems. Huge communication overheads between the terminal and the server and heavy loads of storage space are needed when a large number of data points are uploaded. To this end, an online compression algorithm for positioning data acquisition is proposed, which compresses data by reducing the number of uploaded positioning points. Error threshold can be set according to users' needs. Feature points are extracted to upload real-timely by considering the changes of direction and speed. If necessary, an approximation trajectory can be obtained by using the proposed recovery algorithm based on the feature points on the server. Positioning data in three different travel modes, including walk, non-walk and mixed mode, are acquired to validate the efficiency of the algorithm. The experimental results show that the proposed algorithm can get appropriate compression rate in various road conditions and travel modes, and has better adaptability.

Povzetek: Predstavljen je nov algoritem za zajemanje podatkov o realnem času, uporaben za sisteme za določanje položaja.

1 Introduction

With advances in tracking technologies and the rapid improvement in location-based services (LBS), a larger amount of location data needs to be collected [1]. These positioning data are linked up in chronological order to form a trajectory which contains a sequence of positioning data with latitude, longitude and timestamp. Trajectory can be used for positioning, navigation, providing some location-based services and expressing user geospatial history behaviour. For location acquisition systems, a large number of positioning data not only generate lots of communication overheads but also bring heavy burden for the storage system [2]. A major reason for this problem is that the uploaded data points are far more than necessary points [3]. Therefore, trajectory compression and simplification have received increasing concern.

Trajectory compression can be classified into two categories, offline compression and online compression. In the offline compression, all data points collected (usually on the server) are considered for compression regardless of the data transfer process. In the online compression, data points are processed before they are uploaded, and only feature points selected are transmitted to the server.

For the offline compression, Zhang et al. [2] proposed a trajectory data compression algorithm based on the spatial-temporal characteristic. In order to extract accurately trajectory information, the algorithm calculates the distance standard to judge the feature point according to the three-dimensional spatial-temporal characteristic of GPS data points. Given a trajectory *Traj* and error threshold, the Doulas-Peucker Algorithm [4] constructs a new trajectory *Traj*' by adding points from *Traj* repeatedly until the maximum spatial error of *Traj*' becomes smaller than error threshold. The Douglas-Peucker algorithm has the limitation of ignoring temporal data. Top-Down Time Ratio (TD-TR) [5] overcomes this limitation by using *Synchronized Euclidean Distance (SED)* [6] instead of spatial error. The characteristics of the vehicle trajectory are described in the proposed algorithm [7], in which the advantages and disadvantages of judgment method with point by point and judgment method with multi-point joint are analysed. This algorithm puts forward the description method for trajectory based on turning point judgment method. Chen et al. [8] proposed a trajectory simplification algorithm (TS), which considered both the shape skeleton and the semantic meanings of a GPS trajectory. The heading change degree of a GPS point and the distance between this point and its adjacent neighbours in this algorithm are used to weight the importance of the point. By studying the model of temporal data in vehicle monitoring system, Wang et al. [9] proposed a modelling idea for establishing trajectory version. To reduce storage and improve temporal queries, spatial-temporal cube is segmented to form unit spatial-temporal cube by this modelling idea. With discussing some problems of trajectory data compression about road net, Guo [10] proposed a non-linear compression algorithm of moving objects trajectories based on road net.

The above offline algorithms have played a positive role in reducing amount of positioning points. However, they are only applied to the case that the start and end of the trajectory are clear. That is to say, compression begins only after obtaining all of the points from the input trajectory. That is why communication overheads

between the terminal and the server are huge and burdens of storage space are heavy. In recent years, some online compression algorithms have been proposed.

Similar to Douglas-Peucker, Opening Window algorithm [11] approximates each trajectory using an increasing number of points so that the resulting spatial error is smaller than the bound. This algorithm is a generalized online compression with a little delay, and it works without considering time data. Opening Window Time Ratio (OPW-TR) [5] is an extension to Opening Window which uses SED instead of spatial error. Compared to Opening Window algorithms, OPW-TR has the benefit of taking temporal data into account. However, just like Opening Window algorithm, OPW-TR also has a little delay. Both of them cannot real-timely determine whether current point needs to be compressed. An algorithm called SQUTSH (Spatial QUalIty Simplification Heuristic) [12] uses a priority queue where the priority of each point is defined as an estimate of the error that the removal of that point would introduce. SQUISH compresses each trajectory by removing points of the lowest priority from the priority queue until it achieves the target compression ratio. This algorithm is fast and tends to introduce small errors during compression. However, it cannot compress trajectories while ensuring that the resulting error is within a user-specified bound. It also exhibits relatively large errors when the compression ratio is high [13]. Yin et al. [14] proposed an algorithm and a restraining model for vehicle data stream transmission. It gives a formula for data recovery in the monitoring centre by discussing the determination condition of speed and direction change degree in the algorithm. However, this algorithm is suitable for the integrated system of monitoring and navigation, and must be matched to the map. Wang et al. [15] proposed an improved moving object model based on vehicle monitoring system of moving objects model. This model provides a dynamic time selection algorithm, which adjusts dynamically the time density according to object moving speed. Therefore, the number of communications is reduced effectively. However, this algorithm works regardless of data recovery and the issue that the moving direction affects the accuracy of the model. Han et al. [16] proposed an adaptive algorithm for vehicle GPS monitoring data. The speed function is expressed by using fitting polynomial with constraint based on endpoint speed and mileage. This algorithm considers the case that terminal with and without map, in which the deviation distance is used for denoting the deviation of speed direction. At the same time, data recovery algorithm, algorithm evaluation index and application mode of algorithm are proposed in this algorithm. However, it must be set speed threshold in advance according to the road. In practice, according to the different road, predetermined speed threshold is very difficult.

In this paper, an online compression algorithm for positioning data acquisition systems is proposed, which compresses the data points before transmission from terminal to the server. In this algorithm, the feature points are extracted and uploaded by computing the change of speed and direction based on error threshold set according to the users' requirements, and non-feature points are discarded to reduce unnecessary data transmission and bandwidth occupation.

The remainder of this paper is organized as follows. Section 2 describes our new online compression algorithm for positioning data acquisition in detail. An experimental evaluation of trajectory compression algorithm is provided in Section 3. The paper concludes with future work in Section 4.

2 The online compression algorithm

In order to compress an original trajectory, the key of our algorithm is to pick out feature points .For each point collected in the terminal, the current point will be uploaded and preserved if it is a feature point, else it will be discarded as a non-feature point. If needed, the recovery trajectory that approximated to the original trajectory can be obtained by the recovery algorithm with the feature points on the server. Our algorithm involves feature points, non-feature points and lost points. A feature point is a positioning point that a large change in speed or direction; a non-feature point is a positioning point that is discarded with a small change in speed and direction; a lost point is a positioning point that is lost because the communication delay caused by the positioning signal blind area. Generally, original trajectory is a temporally ordered sequence of positioning points $P = \{p_1, p_2, \ldots, p_n\}$, for $i = 1, 2, \ldots, n$, $p_i = (lng_i, lat_i, t_i)$ where lng_i, lat_i represent the longitude and latitude of the ith point and t_i stands for the time stamp. For a point p_i, $p_i.lng$, $p_i.lat$ and $p_i.t$ denote longitude, latitude and time stamp of p_i respectively. For each positioning point p_i in an original trajectory, recovery point p_i' of p_i is a data point obtained by the recovery algorithm, which represents p_i in the recovery trajectory.

2.1 Algorithms

In the compression phase, error threshold and acquisition interval are set in advance according to accuracy requirements. The initial two collected points (these two points must be the feature points) are directly uploaded to the server without judgment. Starting from the third positioning point, corresponding recovery point is calculated for each point collected currently. The distance between the actual point p_i and its corresponding recovery point p_i' is compared with the error threshold. If larger than the error threshold, the current point is regarded as a feature point and uploaded. If not, the current point is abandoned as a non-feature point (see Fig. 1). Judgment is done for each point, and the last positioning point is uploaded until the end of the acquisition.

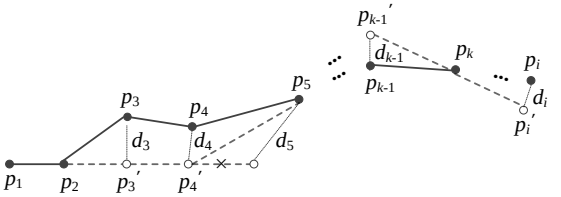


Figure 1: A schematic diagram for the compression process.

As shown in Fig. 1, p_1, p_2, p_3, p_4, p_5, p_{k-1}, p_k and p_i are the original positioning points, each positioning point has a corresponding recovery point. p_1, p_2, p_5 and p_k are feature points, their recovery points are themselves. p_3', p_4' and p_{k-1}' are recovery points of p_3, p_4 and p_{k-1}. p_2 is previous feature points of p_3' and p_1 is previous points of p_2. p_3' is calculated by p_1 and p_2. The speed of p_2 can be concluded by distance and time d-value between p_1 and p_2. Supposing p_2 and p_3' have the same speed. The product of this speed and interval T is the distance of p_3'. Set p_3' on a straight line that p_1, p_2 are located at. According to the coordinate of p_1 and p_2, the coordinate of p_3' can be calculated. In the same way, p_2 is still previous feature points of p_4' and p_5', and they are also calculated by p_1 and p_2. Differently, the distance of p_4' is the product of speed of the p_2 and twice interval T, the distance of p_5' is the product of speed of the p_2 and triple interval T. The distance between recovery point p_i' and actual point p_i is called d_i and d_i is compared with the error threshold D. If $d_i \geq D$, current point is uploaded as a feature point. If not, current point is discarded as non-feature point. d_3, d_4 in Fig. 1 are smaller than D, so p_3, p_4 are discarded. d_5 is not less than D, so p_5 is uploaded as feature point. The recovery point p_i' of current point p_i is calculated though the previous feature points p_k of p_i' and the previous point p_{k-1}' (it can be a feature point or not) of p_k. The speed of p_k is calculated according to p_k and p_{k-1}'. The product of this speed and time d-value between p_i and p_k is the distance of p_i'. Set p_i' in the straight line that p_k, p_{k-1}' are located at. According to the coordinate of p_k, p_{k-1}' and the distance of p_i', the coordinate of p_i' is calculated. In the recovery phase, p_1, p_2, p_5 and p_k are recovered directly as feature points. p_3, p_4 and p_{k-1} are non-feature points. They are recovered to p_3', p_4', p_{k-1}' according to the above process.

In the recovery phase, the first and second feature points received are recovered directly. Determine whether there are non-feature points between the current p_i and previous feature point p_k from the third feature point. If there are non-feature points, they are recovered in order by calculating corresponding recovery points according to p_k. Then recover the current point p_i after all non-feature points between p_i and p_k are recovered. The process cycles until the recovery of the last feature point.

To deal with the problem of lost point, positioning point p_i introduces lost point count (Count) property and it is given with $p_i = (lng_i, lat_i, t_i, c_i)$. The design idea is to add a counter in the terminal. Positioning data are collected at an interval. This interval is consistent for counter and data acquisition. When the time reaches an interval, if the signal is acquired, the counter is "0". If the signal is not acquired, that is the signal loss, counter is added with "1" until the signal is acquired. Fig. 2 shows a schematic diagram for the lost point.

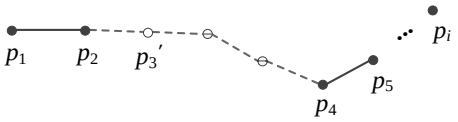


Figure 2: A schematic diagram for the lost point.

As shown in Fig. 2, p_1, p_2 and p_4 are original positioning points. After compression, p_3 is discarded as non-feature point and p_3' is corresponding recovery point of p_3. There are three intervals between p_3 and p_4, and it should have two positioning points between them. However, due to signal loss, these two points are not collected. They are called the lost points. When determining whether p_5 is a feature point or not, p_3' and p_4 are made use for calculation direct without considering the lost points. When recovery of p_3', the lost points are needed to consider. There are four intervals between two feature points p_2 and p_4. It should have three positioning points. Since $C_4 = 2$, that is to say, two positioning point are lost, so just to recover the only non- feature point p_3.

2.2 Algorithm description

The recovery point is used to determine whether the current point is a feature point in the compression phase. The recovery point is the non-feature point recovered in the recovery phase. Therefore, the calculation of recovery point is the key of the algorithm. Making use of the previous feature point p_k of p_i' and the previous point p_{k-1}' of p_k to calculate the recovery point p_i' of current point p_i (if $i = 3$, then $p_k = p_2$, $p_{k-1}' = p_1$).

2.2.1 The compression algorithm

The recovery point of current point is calculated from the third positioning point collected. The actual point has attributes of latitude and longitude. For ease of calculation, latitude and longitude as spherical coordinate can be transformed into rectangular coordinate by the algorithm [17]. That is to say, $p_i = (long_i, lat_i, t_i)$ as spherical coordinate is transformed into $p_i = (x_i, y_i, t_i)$ as rectangular coordinate. For any two points p_i and p_j, the distance of them is $L_j = Dist(p_i, p_j)$ in the trajectory with n points. Accordingly, the distance between p_k and p_{k-1}' is L_k. p_k and p_{k-1}' are the consecutive points and the time d-value of them is $p_k.t - p_{k-1}'.t$ (in the case of signal not loss, the d-value is the interval T). The speed V_i is calculated by Formula (1) and interval T. The distance S_i of p_i' is calculated according to the time d-value of p_i, p_k and Formula (2).

$$L_k = Dist(p_{k-1}', p_k) = \sqrt{(p_k.x - p_{k-1}'.x)^2 + (p_k.y - p_{k-1}'.y)^2}, \quad 2 \le k \le n-1 \tag{1}$$

$$V_i = \frac{L_k}{p_k.t - p_{k-1}'.t} = \frac{\sqrt{(p_k.x - p_{k-1}'.x)^2 + (p_k.y - p_{k-1}'.y)^2}}{p_k.t - p_{k-1}'.t}, \quad 2 \le k \le n\text{-}1, 3 \le i \le n \tag{2}$$

$$S_i = V_i \times (p_i'.t - p_k.t) = \frac{\sqrt{(p_k.x - p_{k-1}'.x)^2 + (p_k.y - p_{k-1}'.y)^2}}{p_k.t - p_{k-1}'.t} \times (p_i'.t - p_k.t), \quad 2 \le k \le n\text{-}1, 3 \le i \le n \tag{3}$$

After calculating the distance of p_i', position coordinate of p_i' is calculated according to rectangular coordinates of p_k and p_{k-1}' and triangle similar properties [18]. Formulas are as follows. Detailed compression process is shown in Algorithm 1.

$$\begin{aligned} p_i'.x &= p_k.x + \frac{s_i \times (p_k.x - p_{k-1}'.x)}{L_k} \\ &= p_k.x + \frac{(p_i.t - p_k.t) \times (p_k.x - p_{k-1}'.x)}{p_k.t - p_{k-1}'.t}, \quad 2 \le k \le n\text{-}1, 3 \le i \le n \end{aligned} \tag{4}$$

$$\begin{aligned} p_i'.y &= p_k.y + \frac{s_i \times (p_k.y - p_{k-1}'.y)}{L_k} \\ &= p_k.y + \frac{(p_i.t - p_k.t) \times (p_k.y - p_{k-1}'.y)}{p_k.t - p_{k-1}'.t}, \quad 2 \le k \le n\text{-}1, 3 \le i \le n \end{aligned} \tag{5}$$

Algorithm 1 for compression

Input: p_i , p_k *and*p_{k-1}' , D and T.
Output: A simplified trajectory *Traj′* with *feature points*.
1. Collect data points and open counter , *Traj′* = *Φ;*
2. Upload p_1 and p_2; // The first and second positioning point needn't decide and are uploaded directly
3. $p_k = p_2$,$p_{k-1}' = p_1$; // p_2 is assigned to p_k, p_1 is assigned to p_{k-1}'
4. **Foreach** p_i ($i \ge 3$); // Cyclic judge for p_i
5.*Transform* $p_i = (lng_i, lat_i, t_i)$ to $p_i = (x_i, y_i, t_i)$; // Spherical coordinate is transformed into rectangular coordinate
6. Calculate S_i; // Calculate the distance of p_i'
7.Calculate (x_i,y_i); // Calculate the rectangular coordinate of p_i'
8.Transform $p_i = (x_i, y_i, t_i)$ to $p_i = (lng_i, lat_i, t_i)$; // Rectangular coordinate is transformed into spherical coordinate
9. *Dist*(p_i, p_i') is d_i; // The distance of p_i and p_i' is d_i
10. Compare d_i and D ; // Error threshold is D
11. If（$d_i \ge D$）, upload p_i ; // If $di \ge D$， upload p_i as a feature point
12. $p_k = p_i$, $p_{k-1}' = p_{i-1}'$; // p_i assigned to p_k， p_{i-1}' assigned to p_{k-1}'
13. Else（$d_i < D$）, ignore p_i; // If $di < D$， discard p_i as a non-feature point
14. **End** foreach; // End of cycle
15. **Return** *Traj′* = {$p_1, p_2, \ldots p_k, \ldots p_n$ }

2.2.3 The recovery algorithm

Determine whether exist the non-feature points between the current point p_i and the previous feature point p_k from the third feature point received. Judgment is that getting the number N_i of intervals based on the time d-value of two points divided by the interval T. By the Formula (6) and the number C_i of lost points p_i, the number M_i of recovery point is calculated. Special attention is that the interval number of two adjacent original positioning points is "1", but there is no recovery point between them. Therefore, the number of

recovery points should extra minus "1". If M_i is not equal"0", it means that it exists recovery points. To recover the M_i recovery points in order. The speed of the recovery point is calculated by Formula (2), and the product of speed and r interval T is the distance G_r of the *r-th* recovery point.

$$N_i = \frac{p_i.t - p_k.t}{T}, \quad 3 \le i \le n, 2 \le k \le n\text{-}1 \tag{6}$$

$$M_i = N_i - C_i - 1, \quad 3 \le i \le n \tag{7}$$

$$G_r = V_i \times r \times T, \quad 3 \le i \le n, 1 \le r \le M_i \tag{8}$$

After calculating the distance of each recovery point, the position coordinate of each recovery point is calculated according to rectangular coordinate of p_k and p_{k-1}', triangle similar properties and Formula (4), (5). Detailed recovery process shown in Algorithm 2.

Algorithm 2 for recovery

Input: Feature points and T.
Output: A recovered trajectory $Traj$ with feature points and recovery points.
1. Receive feature points and $Traj = \Phi$;
2. Recover p_1 and p_2,$Traj = \{p_1, p_2\}$; // The first and second feature points are recovered directly without deciding
3. $p_k = p_2$, $p_{k-1}' = p_1$; // p_2 is assigned to p_k, p_1 is assigned to p_{k-1}'
4. **Foreach** p_i ($i \ge 3$); // Cyclic judge for current feature point p_i
5.Calculate M_i ; // Calculate the number of recovery points between current feature point and previous feature point
6. **Foreach** r ($1 \le r \le M_i$); // Cycle begins from 0 to M_i non-feature points
7.Transform $p_i = (lng_i, lat_i, t_i)$ to $p_i = (x_i, y_i, t_i)$; // Spherical coordinate is transformed into rectangular coordinate
8.Calculate G_r ; // Calculate the distance G_r of current recovery point
9.Calculate (x_i, y_i); // Calculate the rectangular coordinate of current recovery point
10.Transform $p_i = (x_i, y_i, t_i)$ to $p_i = (lng_i, lat_i, t_i)$; // Rectangular coordinate is transformed into spherical coordinate
11. Recover recovery points ; // Recover recovery points
12. Recover p_i ; // Recover current feature point p_i
13. $p_k = p_i$, $p_{k-1}' = p_{i-1}'$; // p_i is assigned to pk, p_{i-1}' is assigned to p_{k-1}'
14.**End** foreach; // End of the cycle
15. **Return** $Traj = \{p_1, p_2, \ldots, p_n\}$;

Compression algorithms can be divided into lossless and lossy compression. Lossless compression enables exact reconstruction of the original data with no information loss. In contrast, lossy compression introduces inaccuracies when compared to the original data. Lossy compression allows the compression process to loss some information. Although we cannot fully recover the original data, but the lost part has a smaller impact on the understanding of the original data, at the same time, in exchange for a smaller compression rate [19]. The primary advantage of lossy compression is that it can often reduce the storage requirements drastically while maintaining an acceptable degree of error [20]. Although all compressed data are recovered by the algorithm proposed in this paper, but it has a certain geographical deviation, so that this algorithm is lossy compression.

3 Experiments

To verify the applicability and performance of the algorithm, a large number of experiments are performed about original data acquisition, data compression and data recovery real-time by *PC* and *GPS* phones.

3.1 Experimental environment

The existing four global satellite navigation system in the world, namely the U.S. GPS system, China Beidou system, the European Galileo system and Russia GLONASS system [21]. GPS system is widely used because of its wide coverage, high positioning accuracy, short positioning time and small location-dependent and other advantages. Positioning data are collected real-timely with Android operating system and GPS positioning system in the experiment. Android includes API libraries that simplify the development related to the device hardware. Android SDK includes location-based services, and determines the current location by GPS.

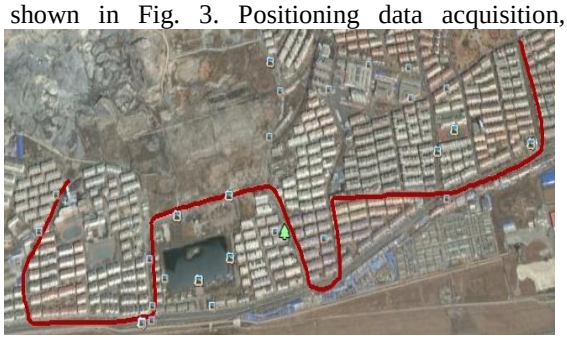
The mobile phone is used as terminal in the experiment, which is regarded as a GPS receiver and a compression processor for positioning data. The PC is used as server, which is used for receiving positioning data compressed, and recovering the original trajectory based on those data.

3.2 Select the experimental route

The actual movement broadly divided into walk, non-walk (car, bicycle, aircraft, ships, etc.) and mixed mode (alternating walk and non-walk) according to different travel modes. According to different speed, it can be broadly divided into 0-30 *km/h* as the low-speed movement (walking, jogging, bicycle, etc.), 30-60 *km/h* as the middle-speed movement (low speed vehicles, etc.) and 60 *km* above as high-speed movement (cars, boats, airplanes of high speed, etc.). Assuming the original trajectory with n positioning points, the purpose of the experiment is to select as few as m feature points. When $m < n$, compression trajectory is constituted with m feature points. Set $r = m / n$ is the compression rate, the r is the smaller the better without changing the original shape of the trajectory of geography. For moving at different speed and different travel modes, it should set a different error threshold. In this paper, the

experiment is given priority to low speed movement, and divided into walk, non-walk and mixed mode for validation. Data is acquired at 2*s* intervals, 5*s* intervals, 10*s* intervals and 20*s* intervals. For each acquisition interval, 5*m*, 10*m*, 15*m*, 20*m* and 25*m* as the error threshold are set for compression.

In this experiment, the actual routes as an example are shown in Fig. 3. Positioning data acquisition, compression and recovery are executed real-timely. Fig. 3*a*) shows a route of non-walk (by bus) and mixed mode (by bus and walk); Fig. 3*b*) shows a route of walk. Due to traffic restrictions, travel by bus is slow. So this experiment is under the condition of low-speed movement.



a) The route of non-walk (by bus) and mixed mode (by bus and walk)

b) The route of walk

Figure 3: A schematic diagram for actual routes.

3.3 Experimental results and analysis

3.3.1 Effects of different parameters on the compression rate

In the actual routes shown in Fig. 3, the compression rate is impacted by the acquisition interval, the error threshold and the travel mode.

Under the conditions of the error threshold are 5*m*, 10*m*, 15*m*, 20*m* and 25*m*, and the travel mode is walk, the effect of different acquisition intervals on compression rate is shown in Fig. 4.

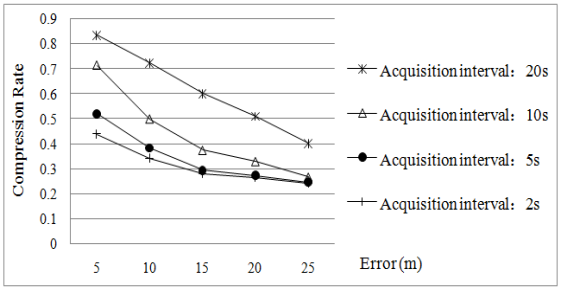


Figure 4: Compression rate of different acquisition intervals.

The Fig. 4 shows that, for the walk of low-speed movement, the shorter the acquisition interval is, the more positioning data are compressed and the smaller the compression rate is. At the same time, we can draw that data compression rate decreases with increasing error threshold. That is to say, the bigger threshold is set, the more data are discarded and the smaller the compression rate is. With the increasing of the threshold, it also brings the risk of losing detailed information.

Under the conditions of the error threshold are 5*m*, 10*m*, 15*m*, 20*m* and 25*m*, and the acquisition interval is 10*s*, the effect of different travel modes on compression rate is shown in Fig. 5.

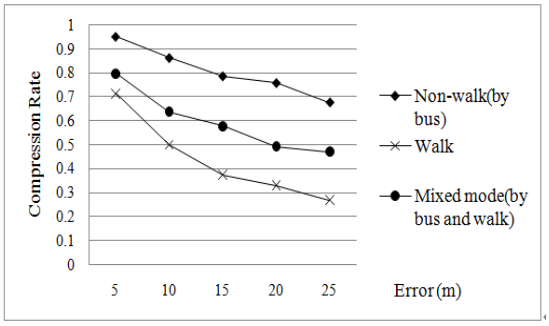


Figure 5: Compression rate of different travel modes

The Fig. 5 shows that for low-speed movement, walk on the relatively straight route without more detours and its speed is more even, so it has the minimum compression rate at the same error threshold. Non-walk is on the route with more detours and it is ongoing to stop and go, at the same time, the change degree of speed is bigger, so it has the maximum compression rate at the same error threshold. Mixed mode includes walk and non-walk, so the compression rate ranges between them.

3.3.2 Effectiveness

The recovery trajectory is compared with the original trajectory to verify the effectiveness of proposed algorithm. The recovery trajectory approximate the original trajectory can be obtained in the case that error threshold is set appropriately, as shown in Fig. 6.

a) Original trajectory b) Compression trajectory c) Recovery trajectory

Figure 6: A schematic diagram for original trajectory, compression trajectory and recovery trajectory.

The Fig. 6 shows a schematic diagram for original trajectory, compression trajectory and recovery trajectory of walk under the conditions of acquisition interval is 10*s* and the error threshold is 10*m*. Fig. 6*a*) shows the original trajectory with 112 points; Fig. 6*b*) shows the compressed trajectory according to the online compression algorithm for positioning data acquisition with 42 points; Fig. 6*c*) shows the recovery trajectory by recovery algorithm with 112 points, the same number as the original data, recovery trajectory approximates original trajectory. That is to say, the compression rate is 37.5%, and denser data are discarded. In addition, the spatial error of points recovered is strictly controlled under the error threshold. The above fact shows that the proposed algorithm is effective in practical applications. The deviation of a recovery trajectory depends on the error threshold predetermined. The larger error threshold is set, the more non-feature points are compressed, the compression rate is smaller, but the deviation between original data and recovered data is larger. Conversely, the smaller error threshold is set, the less non-feature points are compressed, the compression rate is larger, but the deviation between original data and recovered data is smaller, the recovered data are more approximate to the original data.

4 Conclusion

In this paper, an online compression algorithm for positioning data acquisition is proposed based on a straight line approach. Considering the change of speed and direction, the error threshold between the actual point and recovery point is set, as the required accuracy of user, to replace the threshold of speed or direction. As a result, the feature information loss and trajectory error can be significantly reduced. The algorithm is suit for different road conditions and travel modes, so it has a better adaptability than the existing compression algorithm. In the future, one possible approach for improving the compression of positioning data would be to use knowledge of the road network. This approach would require fast detection of deviations from the road network. A key goal in this approach is that enable a smaller compressed representation with low calculating overhead and reduced error.

References

[1] Choi, D.Y. (2007) "Personalized local internet in the location-based mobile web search", *Decision Support Systems*, Vol.43 No.1, pp. 31- 45.

[2] Zhang, D.F. and Zhang, X.M. (2013) "Compression algorithm of GPS trajectories data based on the temporal characteristics", *Journal of Transport Information and Safety*, Vol. 31 No.3, pp. 6-9.

[3] Kaul, S., Gruteser, M., Rai, V. and Kenney, J. (2010) "On predicting and compressing vehicular GPS traces". In: *Proceedings of the 2010 IEEE international conference on communications Workshops (ICC Workshops)*, pp. 1–5.

[4] Douglas, D.H. and Peuker, T.K. (1973) "Algorithms for the reduction of the number of points required to represent a line or its caricature", *Can Cartogr*, Vol.10 No.2, pp. 112–122.

[5] Meratnia, N and de By RA (2004) "Spatiotemporal compression techniques for moving point objects", In: *Proceedings of the 9th international conference on extending database technology (ED BT)*, Crete, Greece, pp. 765–782.

[6] Potamias, M., Patroumpas, K. and Sellis, T. (2006) "Sampling trajectory streams with spatio-temporal criteria", In: *Proceedings of the 18th international conference on scientific and statistical database management (SSD BM)*, pp. 275–284.

[7] Xu, J. and Tian, S.L. (2010) "GPS location data reducing based on turning point judgment method", *Computer Engineering*, Vol.36 No.7, pp. 268 -269 +272.

[8] Chen, Y.K., Jiang, K., Zheng, Y., Li, C. and Yu, N. (2009) "Trajectory simplification method for Location-Based Social Networking Services", *ACM GIS workshop on Location-based social networking services*, pp. 33-40.

[9] Wang, W.J., Weng, J.N. and Fan, K. (2006) "Design and implementation of spatial-temporal data model in vehicle monitor system", *Computer Engineering and Design*, Vol.27 No.6, pp. 1042 - 1044 +1051.

[10] Guo, Z.Z. (2008) "Compression of moving objects trajectories based on road net", *Computer and Modernization*, No.2.

[11] Keogh, E.J., Chu, S., Hart, D. and Pazzani, M.J. (2001) "An online algorithm for segmenting time series", In: *Proceedings of the 2001 IEEE international conference on data mining (ICDM)*, California, USA, pp. 289–296.

[12] Muckell, J., Hwang, J.H., Patil,V., Lawson, C.T., Ping, F. and Ravi, S.S., (2011) "SQUISH: an online approach for GPS trajectory compression", In: *Proceedings of the 2nd international conference on computing for geospatial research and applications (COM. Geo)*, New York, USA, pp. 13.1–13.8.

[13] Muckell, J., Olsen Jr, P.W., Hwang, J.H., Lawson, C.T., and Ravi, S.S., (2014) "Compression of trajectory data: a comprehensive evaluation and new approach", In: *Geoinformatica*, Vol.18, pp. 435-460.

[14] Yin, J.Z., He F.Q., Li Q.Q. and Xu, L. (2008) "Data stream transmission control algorithm in navigation and monitoring integrated system", *Geomatics and Information Science of Wuhan University*, Vol. 33 No. 9, pp. 900-903.

[15] Wang, S.S. and Liu, D.Y. (2004) "Moving object model based on vehicle monitor system", *Chinese Journal of Scientific Instrument*, Vol. 25 No.4, pp. 509-511 +524.

[16] Han, X.G., Yan, L. and Lu, W. (2011) "Adaptive acquisition algorithm for vehicle GPS monitoring data", *Geography and Geo-Information Science*, Vol. 27 No.3, pp. 12-16.

[17] Yang, P. (2009) "An easy way to convert GPS coordinates in EXCEL", *Fortune China Science and Technology* , No. 04.

[18] Chen, Y.M., Feng, H.L and Dai, Q.H. (2010) "The research of Delta grey correlation based on similar triangle", *Statistics and Decision*, No. 23, pp. 16-18.

[19] Yang, H.Y. and Zhang, Y. (2002) "Design and implementation algorithm of the GPS positioning data compression", *Application of Electronic Technique*, No.12, pp. 29-32.

[20] Muckell, J., Hwang, J.H., Lawson, C.T. and Ravi, S.S., (2010) "Algorithms for compressing GPS trajectory data: an empirical evaluation", In: *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems* (*ACM - GIS*), pp. 402–405.

[21] Liu, M.S. (2006) "Global positioning system and its application(1)—history of development for navigation and positioning technology", *China measurement technology*, Vol. 32 No.5, pp. 1 -7.

Semantic Annotations for Workflow Interoperability

Hamri Salah
Laboratory LIRE, University of Constantine 2, Algeria
E-mail: hamri.salah@yahoo.fr

Boudjlida Nacer
University of Nancy 1, LORIA
UHP, France
E-mail: Nacer.Boudjlida@loria.fr

Boufaida Mahmoud
Laboratory LIRE, University of Constantine 2, Algeria
E-mail: mboufaida@umc.edu.dz

Keywords: workflow model, interoperability, semantic annotations, ontologies, common workflow ontology, common process semantic annotation model

Received: July 13, 2014

To support the interoperability or the cooperation between different partners, various approaches and technological solutions were proposed, which converge directly to the adoption of standards. Consequently, the semantic aspect is not correctly addressed by today's interoperability solutions that focus mainly on the syntactical and technical level. Indeed, addressing the semantic aspect at conceptual level will provide more flexibility to the cooperation. Accordingly, in this paper, we propose an agnostic approach for the interoperability of Workflow models (or business process), which is used in a homogeneous or in a heterogeneous context. In a homogeneous context, lexical and structural annotations are attached to models. Contrary, in a heterogeneous context, we introduce a common semantic annotation structure for annotating the models at different levels: 1) meta-models, 2) models content, 3) models profiles and goals for semantic discovery purposes, 4) at levels of basic aspects of models such as the informational type. Common ontologies, including: Workflow ontology, domain specific ontology, profiles ontology, goals ontology and a set of ontologies related to these aspects, are used to achieve semantic interoperability. One of the advantages of this proposal is its flexibility and its openness since we take an agnostic approach to ontology representation languages (such as OWL-S or WSDL-S)

Povzetek: Predstavljena je nova metoda označevanja za skupno uporabnost delovnega toka.

1 Introduction

To enable interoperability between Workflow models, the WfMC (Workflow Management Coalition) defined a canonical model [8] called XPDL (XML - Process Language Definition) as a language for process interchange and Wf-XML [9] as interoperability protocol between Workflow engines. The Business Process Management (BPM) have mainly focused on Web service technology and came up with a multitude of Web service composition languages standards such as ebXML [3], BPEL4WS (Business Process Execution Language for Web Services) [7]. However, these languages are based on XML and unfortunately lack a semantic description of concepts that enable business process to exchange with a common understanding.

To address the need of semantics, different languages have been proposed such as OWL-S (Ontology Web Language for services) [10], WSMO (Web Service Modeling Ontology) [11], among others. Unfortunately, the problem of the interoperability ever remains at technical level using these ontology representation languages.

To fulfil the need of an independent way of describing semantics to Web services, some authors [13], [14], [15], [16] have developed an approach based on MDA (Model Driven Architecture) [6] for generating OWL-S descriptions. As such, their approach relies upon the use of the OWL-S language, whereas in [4], the authors developed an approach that focuses on WSMO language. Thus, the semantic aspect is not correctly addressed in a conceptual manner.

For bringing semantics to process models at conceptual level (without looking any technology and associated languages), some authors [17] [18], [19], [20], [39], [34] have proposed the use of the semantic annotations by adding metadata and using a set of ontologies to describe the semantics of information in a

heterogeneity context. However, their approach doesn't consider the model aspects while they are essential and intrinsically related to any process model such as the informational or the organizational type. As argued in [43], the purpose of a Workflow model (or business process) is to be executed. Therefore, it is natural for us to refer to its different aspects for execution and to its main objective for which it has been created.

In addition to these works, many approaches have been conducted, with different points by researchers, such as [45], [46], [47], [48], [49] leading to various forms of semantic annotations for process models. However, the authors deal with the problem of semantic heterogeneity of models independently of the context of interoperability where we should align the different terminology that might be used in models and share a conceptualization of the concepts typically employed in the most process meta-models (or process modeling languages). There is no common Workflow ontology (or common process ontology) that enables a common understanding of the models between the actors that wish to interoperate. Additionally, the model aspects are not considered in these approaches.

Moreover, the authors have disregarded the homogeneous context where sometimes the interoperating actors find difficulties for better interpreting their models. These difficulties are caused in general by the unambiguous terms used in the Workflow domain (or business process area).

Consequently, to enhance semantic interoperability in a homogeneous context, we introduce complementary annotations helpful the cooperating actors for expliciting the meaning of the models and for easing their exchanged within that context.

This motivates our interest in developing two conceptual approaches that allow actors to cooperate in both homogeneous and heterogeneous environments.

In the first approach, we suggest the use of lexical and structural annotations for annotating the models within a homogeneous context. Then, in the second approach, to tackle the heterogeneous semantics of distributed process models, we propose an ontology based semantic annotation approach based on building a common process semantic annotation model (CPSAM) for annotating the models at different levels: 1) meta-models, 2) models content, 3) models profiles and goals for semantic discovery purposes, 4) models aspects such as the informational or the organizational type.

The contribution of this paper is as follows. First, we propose an approach for Workflow interoperability that enables the cooperating actors to better interpret the meaning of the exchanged models in a homogeneous context. Lexical and structural annotations are used within that context for annotating the models. Second, to achieve semantic interoperability in a heterogeneity context, a common process semantic annotation model is developed and presented formally. This explicitly defines all necessary annotation elements. We introduce in that context, a set of common ontologies, including: Workflow ontology, domain specific ontology, etc. Third, we define the mapping rules for the meta-models and the model annotation.

The remainder of this paper is structured as follows. In the next section, we describe related work. Then, to achieve semantic interoperability of Workflow models in a homogeneous and heterogeneous context, we propose in Section 3, the different types of annotations that can be attached to models. In section 4, we build a common process semantic annotation model (CPSAM) whose concepts are extracted from common and shared ontological concepts. Section 5 introduces the Supply Chain Operations Reference (SCOR) [36] model as example of reference domain ontology for annotating the models content. Then, we formalize CPSAM in Section 6. For semantic discovery purposes of these models, we enrich in Section 7, the CPSAM with semantic annotations of profiles and goals. Section 8 complements the CPSAM with another type of annotations, which is related to basic aspects of models. Finally, we conclude this paper and outline our future work.

2 Related work

The Web services community has proposed different Web services composition languages such as BPEL4WS [7], WSFL (Web Service Flow Language) [5], ebXML [3] for the interoperability of business process. However, these languages lack a semantic description of concepts for a common understanding of models.

To provide semantic descriptions to Web services, some authors have investigated the area of semantic Web services and have proposed different languages such as OWL-S (Ontology Web Language for services) [10], WSMO (Web Service Modelling Ontology) [11], WSDL-S (Web Service Description Language Semantics) [12], among others. However, the authors rely on the use of these ontology languages to describe the semantics of Web services. Thus, they deal with the semantic aspect at technical level using technologies (tools, existing platforms, languages, etc.) for implementing their approach.

To solve the problem of the semantics of Web services in any independent way of these ontology representation languages, some authors [13], [14], [15], [16] have proposed to create OWL-S descriptions using the MDA approach. Their purpose is to allow a developer to focus on creation of semantic Web services and associated OW L-S specifications via the development of a standard UML model. By using MDA approach, the technique facilitates the creation of descriptions of semantic concepts while hiding the syntactic details associated with creating OWL-S specifications. As it was shown previously in [1], [2], we have proposed a way of addressing the challenges of semantic description of Workflow models (or process models) using the MDA standards and in particular the Ontology Definition Meta-model (ODM) for ontology modeling. However, such approaches are usually applied in an industrial context. Hence, the problem of the semantics of Web services is not correctly addressed in a conceptual way.

In business process area, to specify the semantics of process models, some authors [17], [18], [19], [20], [39], [34] have proposed the use of semantic annotations for achieving semantic interoperability at conceptual level. Their approach focuses on the semantic heterogeneity of the process models on both model and meta-model levels. We see our work as an extension to the existing efforts, because the approach discusses these efforts in the context of interoperability of business process from the perspective of model and meta-model levels. However, the authors omit the basic aspects, which are related to models for execution. Furthermore, they don't consider the homogenous context where the cooperating actors need sometimes additional annotations helpful for a common interpretation of the models within that context.

Similarly, there are other approaches of semantic interoperability such as [45], [46], [47], [48], [49]. Their purpose is to provide semantic annotations for process flows, as well as implemented research prototypes with similar or overlapping capabilities. However, the authors are unaware of context of interoperability where we need a common Workflow ontology (or common process ontology) that should be shared between the actors that wish to interoperate. Furthermore, the authors generally deal with semantic heterogeneity on modeling language level, and omit the meta-model level, which is essential in a context of interoperability. Indeed, within that context, we should address the semantic heterogeneity problem not only on model level or on language such as PSL, but on meta-model level and use one or several ontologies that provide a representation of conceptualization of concepts typically used in the most process modeling languages. Thus, we can facilitate interoperability.

The common factor of these approaches is omitting the most important aspects of business processes, which can be modeled and be investigated independently of each other. As argued in [43], the purpose of any Workflow model or business process is to be executed.

Thus, we must take into account these aspects. Among them, we cite those which are generally considered as essential such as the informational, the organizational, the functional, the behavioural or the resources aspect.

Also, recent works [40], [41] [42], affirm that annotating business processes with a context-based process semantic annotation model facilitates searching models, navigating the repository and enhance understandability of process models. The annotation model consists of the following annotation elements: *process type*, *process area*, *resource*, *actor*, *organizational level*, *process phase*, *process relationship*, *business context*, and *goal.*

However, this annotation model is composed of elements derived from concepts which were elicited (from literature) and validated through an empirical study. Hence, there is no formal semantics of concepts and no reference to one or several ontologies. Indeed, having ontologies will provide a representation of a shared conceptualization and a common understanding of process models in concise and consensual manners for the cooperating partners. Thus, the approach we adopt is based on ontologies and semantic metadata as mediators for reconciling the semantic heterogeneity of process modeling concepts.

Finally, a look at these current approaches has shown, that is a large emphasis on a heterogeneity context and less work in homogeneity context. Indeed, in a real business cooperation and in a homogenous context, the collaborating actors need sometimes annotations when difficulties arise in the interpretation and the understanding of the models.

Accordingly, we address in this paper the semantic heterogeneity problem on both homogenous and heterogeneous contexts. We use ontologies to relate concepts across different modeling languages, as well as to align domain specific terminology used in modeling languages.

Our purpose is therefore to propose a conceptual semantic annotation framework for interoperability of Workflow models (or business process) within or across enterprises. For this purpose, we propose two approaches, which are used in a homogeneous and in a heterogeneous context. In a homogenous context, we attached to models: 1) lexical annotations with reference to lexical data base, 2) structural annotations using only the UML notation. Contrary, in a heterogeneous context, the proposed approach is based on building a common process semantic annotation model (CPSAM) that contains all the common concepts that are shared between the actors that wish to interoperate. In that context, we refer to a set of common ontologies for annotating the models at levels of meta-models and models. To enhance interoperability and to enable a common interpretation and understanding of the models without any ambiguity; we extend CPSAM with annotations of basic aspects of models (such as the informational or the organizational aspect). To enhance reuse of models in their specific projects and especially for profiles or goals-oriented queries, we complement CPSAM with profiles and goals annotations.

One of the advantages of this proposal is its flexibility and its openness since we take an agnostic approach to ontology representation languages (such as OWL-S or WSDL-S). In this way, the cooperating actors can annotate their models (or models fragments) according to their preferred ontology representation language.

3 Annotations for workflow interoperability

In this section, we propose the types of annotations that we feel necessary to achieve semantic interoperability of Workflow (or process) models in a conceptual way. However, two contexts can occur between two interoperating actors: *i)* within a homogenous context in intra-enterprise cooperation, *ii)* within a heterogeneous context in inter-enterprise cooperation.

3.1 Annotations in a homogeneous context

The semantic interoperability within a homogeneous environment does not really constitute a crucial problem for the cooperative companies. Nevertheless, to avoid ambiguities of interpretations and having a good comprehension of the models during the cooperation, it is preferable to annotate them.

In a homogenous context, fitting the models with lexical and structural annotations using the UML notation with reference to same ontologies are sufficient for the actors to better interpret the received models.

In this environment, the cooperative partners (or actors) know each other, share the same field of knowledge, (i.e., even area of reference). Therefore, they must agree on the semantic description of the annotations as well as the various types of annotations, which can be thus elaborate together. Therefore, the reference domain ontology is known and two actors exchange their models *M1* and *M2*, using the same notations for their models as well as the same process ontology *O1* or *O2*. This means that *MM1* and *O1* are same as *MM2* and *O2*, respectively or, equivalently. In this context, lexical annotations described by an actor *Act1* for the model *M1* suffice for the actor *Act2* to interpret the received model. Moreover, to enable consistently understand the meaning of models basic aspects without ambiguity, the interoperating actors refer to same ontologies corresponding to each aspect.

In this homogeneous context, we propose two types of annotations:

- *Lexical/Terminological annotations:* By such annotation, terms are attached to a lexical base data, which make explicit the meaning of models. The definition of terms may be provided by the available ontologies, by the modeling language, as well as by WordNet [21] or by Wikipedia [22]. In addition to this kind annotation, we use a common glossary [23], which is established by the WfMC (Workflow Management Coalition) [24] and contains a description and common terminology for the basic concepts embodied within a process definition.

- *Structural annotations:* that consist in attached concepts from domain ontology to models elements such operations and attributes using the UML notation. The figure 1 shows an example of structural annotations of a purchase order shipping activity (a model fragment) where the elements of the activity refer to a set of ontologies (purchase order, shipment, bank and finances).

Although, we are in a homogeneous context (i.e., same domain ontology (for instance, an automobile area)), we can distinguish two situations that can occur between two actors that wish to interoperate within that context:

Situation 1: When the interoperating actors *Act1* and *Act2* use the same ontologies: This means that is a same understanding that may rely on the use of one or several ontologies, which are identical. These ontologies concern the process models and their basic aspects. In this situation, the annotator annotates the process models *M1* and *M2*, using only the lexical annotations that express the natural meaning of terms. The definitions of the terms may be provided by some lexicons or agreed terminology (like those by WordNet [21], by Wikipedia [22] or by the WfMC's glossary [23], which is established by the organization Workflow [24]).

Situation 2: When the interoperating actors *Act1* and *Act2* don't agree on the same ontologies: This means that each actor use its proper ontology. Thus, *Act1* annotates his model with reference to the ontology *O2*, which is used by *Act2*, i.e., the *O2* concepts are used as metadata to annotate the model M1. Also, *Act2* uses the *O1* concepts as metadata to annotate the model *M2*.

However, in a heterogeneous context, semantics interoperability constitutes a difficult problem in meaning understanding and sharing models in inter-enterprise cooperation. Therefore, we must deal with the semantic heterogeneity of Workflow meta-models and models within that context.

3.2 Annotations in a heterogeneous context

In a heterogeneous context, the cooperating actors may use different notations for their models (for example, *M1* is notated according to the meta-model *MM1* and *M2* to the meta-model *MM2*) and they may refer to different ontologies. For example, Act1 uses an ontology *O1*, while *Act2* uses an ontology O2 different from *O1*.

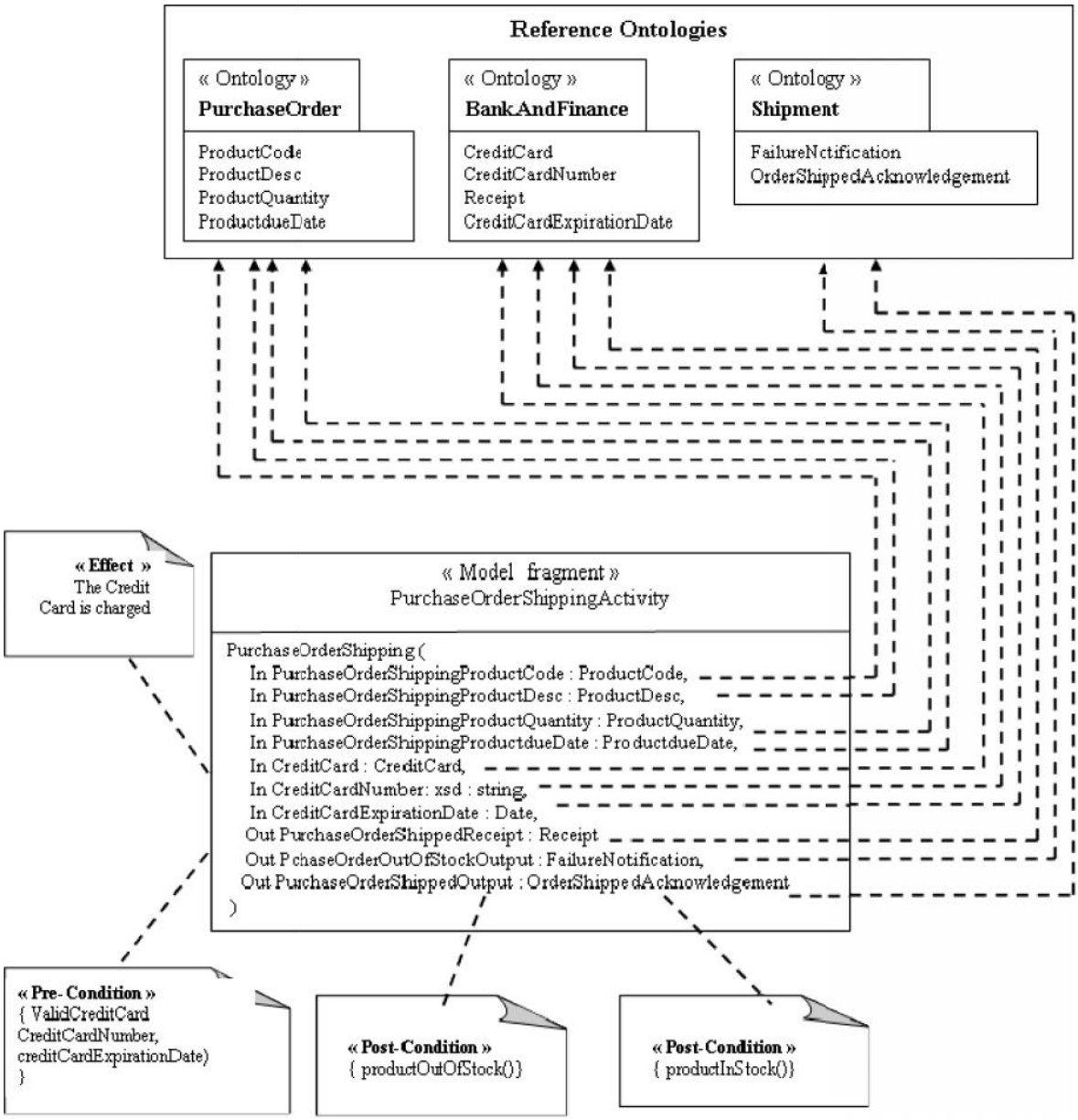


Figure 1: Structural annotations of a model fragment (purchase order-shipping activity).

In order to achieve semantic interoperability of process models within that context, a common understanding of models representation is needed when searching process models. Therefore, we use common ontologies to relate concepts across different modeling languages, as well as to align domain specific terminology used in models. Thus, we annotate the models with reference to a set of common ontologies that are consensually, accept and shared between the interoperating actors.

The proposed approach is therefore an ontology based semantic annotation approach. It relies on the use of several common ontologies, including: Workflow ontology, domain specific ontology, profiles ontology, goals ontology and a set of ontologies related to different basic aspects, which are intrinsically related to process models such as the informational type. For annotating the various models, we define a semantic annotation process (figure 2) based on the following steps:

1. Elaboration of a common process semantic annotation model (CPSAM): In this step, we build a common process semantic annotation model whose concepts derived from a common Workflow ontology (CWO), reference domain ontology and a thesaurus (as a form of ontology).

2. Meta-model annotation: It consists in annotating the concepts of different meta-models of Workflow (such as XPDL or ebXML) by the concepts of the common Workflow Ontology (CWO). The CWO concepts are then used as metadata to annotate the various semantics of concepts of meta-models, i.e. the CWO

concepts will take place the corresponding meta-models concepts to describe the processes.

3. Model annotation: Once, the concepts of meta-models are annotated by the CWO concepts, (i.e. the process models are described by the CWO concepts), we annotate the models content with reference to a domain specific and thesaurus.

4. Profiles and goals annotation: For semantic discovery purposes and especially for profiles or goals-oriented queries, and reuse models in their specific projects, we annotate them with reference to profiles and goals ontology.

5. Annotation of basic aspects: For better interpreting the meaning of the exchanged models without ambiguity, we annotate the models with reference to a set of common ontologies, which are related to the informational, the organizational, the behavioural, the functional or the resources aspect.

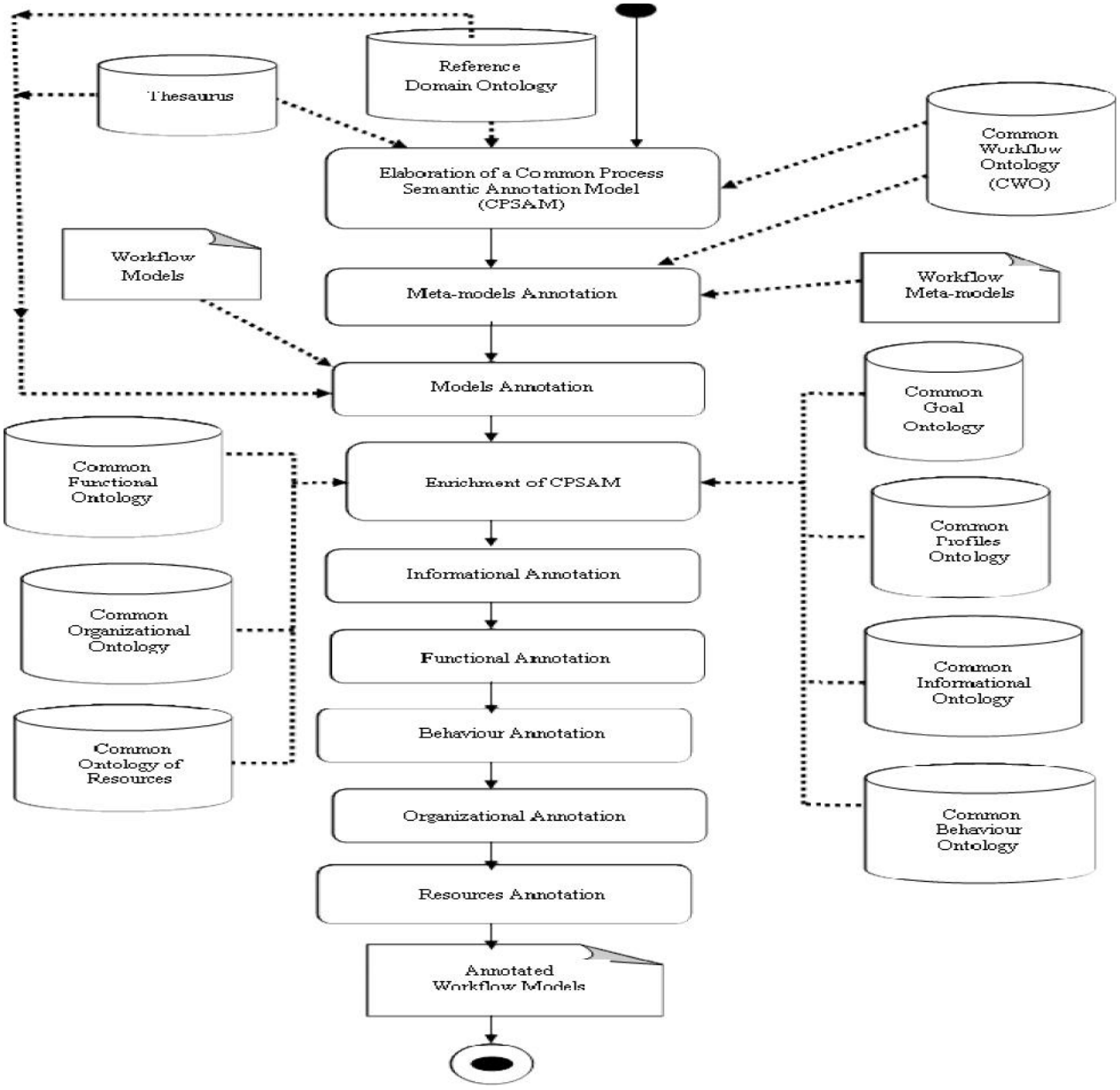


Figure 2: Semantic Annotation Process of Workflow Models.

4 Common process semantic annotation model

To support the various semantic annotations, we define a common annotation scheme (figure 3) dedicated to actors that wish to cooperate. This scheme represents a common process semantic annotation model (CPSAM) that contains all the common and shared ontological concepts between the interoperating actors. These concepts are extracted from a set of common ontologies: Workflow ontology, domain specific ontology, etc.

Following our semantic annotation process as described above, we firstly annotate the semantics of meta-models of process modeling languages. Next, we annotate the models content, then their profiles, their goals and finally, their basic aspects.

4.1 4.1 Meta-model annotation

In the meta-model annotation, we use common Workflow ontology (CWO) as metadata to annotate the semantics of concepts of the different meta-models used in the Workflow domain (or business process). Therefore, we first introduce our common Workflow ontology, and then we describe the way of annotating the semantics of meta-models of process modeling languages.

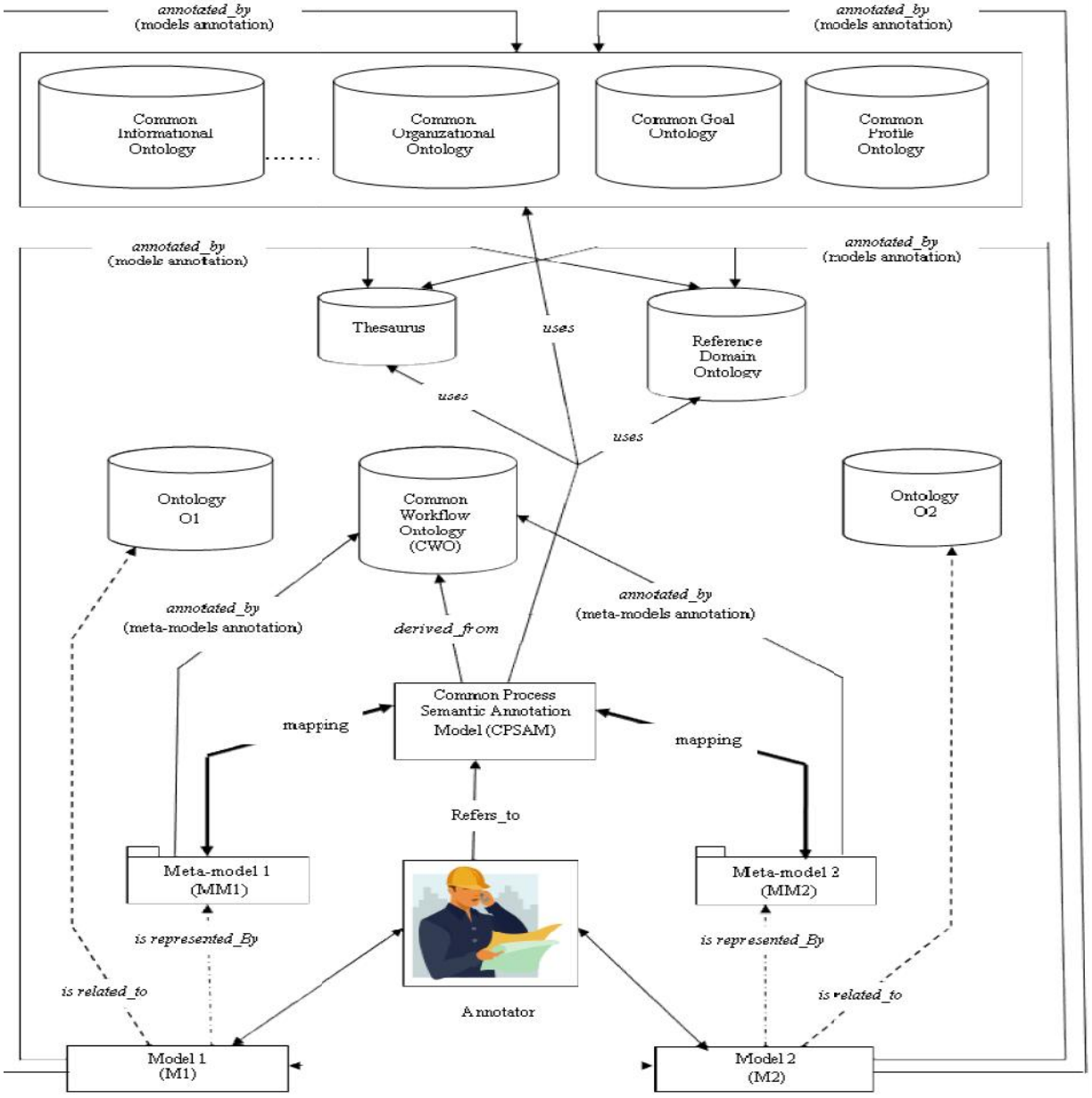


Figure 3: Common Annotation Scheme.

4.1.1 Common workflow ontology

The common Workflow ontology (CWO) is used to align the heterogeneous meta-models of process models. CWO is implemented as an ontology using the Protégé tool [25]. In previous works [1], [2], we have developed an MDA approach for building an OWL ontology for Workflow models according to the investigation of some process modeling languages including ebXML, WSFL, XLANG, BPML, UML, XPDL. We have compared the proposed concepts and have aligned them up according to their objectives as defined by their designers. Then, we have extracted a core of basic concepts that is common to the set of Workflow models. Table 1 shows the alignment of concepts of some process meta-models.

The common Workflow ontology is then derived from a common meta-model (figure 4) that includes the following concepts which are usually modeled as modelling constructs in most process modeling languages : *Wf-Process, Wf-Step, Wf-Activity, Wf-Task, Wf-Transition, Wf-Resource, Wf-Artifact, Wf-Role, Wf-ManualTask, Wf-AutomatedTask*. Compared with our previous works [1], [2], this meta-model is updated here by refining *i)* the concept of Wf-Activity with the relationships: 'kind_of', and 'phase_of' *ii)* the concept of Wf-Actor with the relationships: 'member_of', 'subClass_of' and 'instance_of' *iii)* the concept of Wf-Role with the relationships: 'member_of', subClass_of' and 'instance_of'. These relationships are used to link the concepts in models and concepts in ontologies for the model annotation.

4.1.2 Mapping rules in meta-model annotation

The annotation of certain meta-model has to be done manually by experts who know the process modeling language to be annotated. The procedure of a meta-model annotation is in fact to set mapping rules between the CWO concepts and process modeling language constructs or meta-model concepts.

The mapping rules consist of both one-to-one and one to many correspondences between CWO and meta-model concepts. There may be more complicated cases: a correspondence between a CWO concept and a combination of some meta-model concepts. To define the mapping rules for different cases, we categorize only two of modeling constructs: Atomic Concept, Enumerated Concept. Each concept is an Atomic Concept or an Enumerated Concept. This one is an enumeration set of several Atomic Concepts.

- **Mapping rules**

To establish the correspondence between the concepts of meta-models and CWO concepts, we define two types of mapping:

Workflow Concepts	Concepts of Common Meta-Model (CMM)	XPDL Concepts	ebXML Concepts	WSFL Concepts	XLANG Concepts	BPML Concepts	UML Concepts
Process	Wf-Process	Workflow Process	Multiparty Collaboration Binary Collaboration	FlowModel	Process	Process Abstract	ActivityGraph
Activity	Wf-Activity Wf-Step Wf-Task	Workflow Activity	Business Activity Collaboration Activity Business Transaction Activity	Activity	ActionBase	Activity and its speciali-zations	ActionState SubactivityState FinalState PseudoState
Resource	Wf-Resource	Workflow Application	Business Transaction	---	---	Participant	---
Transition	Wf-Transition	Transition Information	Transition Join Fork	ControlLink Join	Sequence While All	All Sequence Foreach	Transition
Object	Wf-Artifact Wf-Data	Relevant Data	Business Document	DataLink Operation Message	Service Operation CorrelationSetDecl	Message	Classifier ClassifierInState
Role	Wf-Role	Workflow Participant	Business PartnerRole	ServiceProvider	---	Participant	---
Actor	Wf-Actor	Workflow Participant	Autorized Role	---	---	Participant	---

Table 1: Alignment of concepts of some process meta-models.

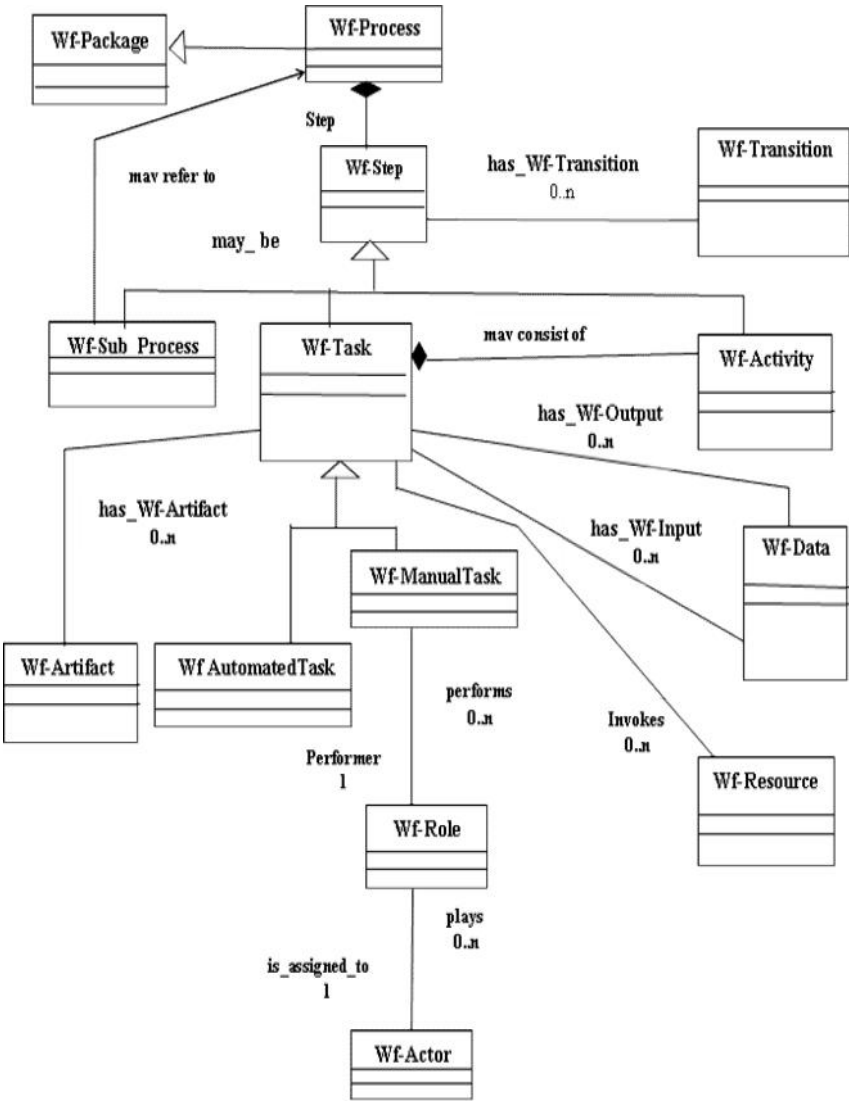


Figure 4: Common Workflow Meta-model.

- ✓ One-to-one mapping: a CWO concept (e.g. CWO:Wf-transition) is referred by an Atomic Concept (e.g. XPDL:TransitionInformation);

- ✓ One-to-many mapping: a CWO concept (e.g. CWO: Wf-transition) can be referred respectively by several concepts (e.g. ebXML: Transition, ebXML: Join and ebXML: Fork) which are enumerated in an Enumerated Concept.

However, these mapping rules may be more complicate: a correspondence between a CWO atomic and a composed concept of one meta-model or between a CWO composed concept and another composed concept of one meta-model. These mapping rules are codified in XML Schemas.

Once the mapping rules are defined for the meta-models, their process models are then be described by the CWO concepts, i.e. the CWO concepts are used as metadata to annotate process semantics. We call the process models described by the CWO metadata as CWO-annotated process models.

4.2 4.2 Model annotation

Based on the meta-models, the CWO concepts will take place the corresponding process modeling constructs to describe the processes. The models (contents) are instances of meta-models and those instances usually describe certain domains. The representations of domains are often various due to diverse uses of terminology and conceptualization, resulting in semantic heterogeneity of models contents. Domain ontologies are agreed as standard representations and semantic definitions of domain concepts by annotation users. Semantic heterogeneity of models (contents) can be reconciled by referencing ontological concepts represented in domain ontologies. Therefore, we use domain ontologies to annotate the models contents and thesaurus as a type of ontology for annotating the used terms to help the cooperating actors for expliciting the meaning of the models.

The annotation method of models is to build relationships between models contents and domain ontologies. The model annotation is therefore to map the

concepts defined in the models to those defined in the domain specific ontology. For this, different mapping methods can be used.

- **Mapping rules in model annotation**

Different mapping strategies can be used between concepts in the models and the domain specific ontology. They can be simple rules applied in meta-model annotation by referring specific model content in modeling constructs to corresponding domain concepts. More complicated mappings can be defined through refined relationships between concepts used in models and concepts defined in domain ontology. The mapping rules are applied using two relationships of mapping:

- ✓ **Simple relationships**: It is a simple mapping by reference. The relationship of mapping can be defined as one type 'refers to'. It assumes that almost all concepts in the model have equal or approximately equal concepts in the ontology. We have adopted such mapping strategy in the meta-model annotation to build the correspondences of concepts between meta-models concepts and the CWO concepts. The strategy of simple reference is easy to apply to map the concepts.

- ✓ **Refined relationships:** The concepts used in process models are variously defined initially for different projects. Therefore, it might be difficult to find equally defined concepts in the domain specific ontology for process models. However, they are still within one domain, there must be some relationships between concepts in models and concepts in ontology. We define some refined relationships to link the concepts between models and ontologies for the model annotation.

The models that are related to domain information are usually artifacts, actors and activities. These concepts will be annotated by domain specific ontology concepts. For this purpose, we use the following relationships for the model annotation.

- The relations: *'kind_of' and 'phase_of'* are used for annotating activities with domain ontology (i.e. relating an activity with concepts defined in domain ontology).

- The relations: *'member_of', 'subClass_of' and 'instance_of*" are used to annotate relationships between roles or actors defined in a model and concepts defined in domain ontology.

- The relations: '*same_as' and 'equivalent_name*' denote the relationship of synonym. They are used respectively, for linking an activity with concepts defined in domain ontology (concept level) and with thesaurus (terminology level).

5 SCOR as example of reference domain ontology

Checking the literatures of specific domain ontologies in the field of business process, we have found the SCOR model as a process reference model. It provides the process templates and standards of logistics process. It is just referenced as an example of domain ontology including the domain activities and the domain goals. Its role is to facilitate the model annotation and the goal annotation.

The Supply Chain Operations Reference (SCOR) [36] model is the product of Supply Chain Council (SCC). It is a process reference model that has been developed and endorsed by the SCC as the cross-industry standard diagnostic tool for supply-chain management. It is has been therefore developed to describe the standard business activities associated with all phases of satisfying a customer's demand.

There are three level process details in the reference model. The top level defines the scope and content of SCOR. Five process types: Plan, Source, Make, Deliver and Return are defined at this level. The second level is the configuration level defining the core "process categories". The third level is the process element level, decomposing the process categories into the process elements. The process element level consists of process element definitions, process element information inputs, and outputs, process performance metrics, best practices, system capabilities required to support best practices, and systems/tools.

The level 3 process elements provide enough details of references for domain activities. In this paper, we model domain ontology concepts based on the SCOR process elements at level 3 for model annotation purposes. The SCOR ontology is formalized in OWL. To organize those concepts, we categorize domain concepts with the concepts (Wf-Activity, Wf-Artifact, Wf-Role, and Wf-Actor) defined in CWO (Common Workflow Ontology). The purpose of the categories is to establish the mapping relationship between a CPSAM model and the reference ontology when annotation. The concepts are modeled as OWL Classes, and they are organized by a subsumption hierarchy.

For annotating the models, we apply, in our case,"S1 Source Stocked Product" model as reference domain ontology for the item receiving process. Each process element is a task ontology concept, which is referenced by the concept Wf-Activity in the CPSAM. For example, two Workflow models are both about purchase order process domain. In one model, an activity called ' Create Order', while in another model, is a called 'Get Order Data'. To enable a same understanding for these activities, we should annotate them by the same concept of the activity SCOR ontology, for instance, 'phase_of Wf-Activity: Schedule Product Deliveries'. That means that are regarded as a phase of the activity ontology 'Schedule Product Deliveries'. Another example, the activities: ' Check Items' and 'Verify Order' are annotated with the activity SCOR ontology 'Verify Product', i.e,

'kind_of Wf-Activity: Verify Product'. For further details on SCOR, we refer the reader to [36].

After the meta-model annotation, the models are then described as a CWO annotated process models. Therefore, the annotation of models is done directly in the CWO annotated process models instead of original models. Thereby, we can formalize the CWO annotated process models in the common process semantic annotation model (CPSAM).

6 Formalization of a common process semantic annotation model

In this section, we build a common process semantic annotation model (CPSAM), which is based on the CWO concepts to describe a process model. The constructs of CPSAM relate some common constructs often used in most process modeling languages and present the main semantics of process information in the models. We formalize CPSAM as follows:

CPSAM = (AV, AR, AC, AF, AI, AO, AT, RS DO).

Where AV is a set of activities composing a process, AR is a set of roles interacting with a process; AF is a set of artifacts participating in a process. AC is a set of actors (agents, users) participating in the process execution. AI is a set of parameters of inputs of an activity. AO is a set of parameters of outputs of an activity. AT is a set of transition conditions of an activity. RS is a set of resources that are invoked during the execution process. DO is a subset of domain ontology concepts.

An activity is a model fragment in CWO, considered as a step in a process and may be a sub-process or atomic activity. Therefore an annotated activity AVi is described as follows:

AVi = (id, name, same_as, equivalent_name, has_Wf-Role, has_Wf-Actor, has_ has_Wf-Artifact, has_ has_Wf-Input, has_ has_Wf-Output, has_Wf-Transition, has_Wf-Resource, kind_of, phase_of)

Each element in CPSAM has id and name to uniquely identify the element. '*Same_as'*, '*kind_of*', and '*phase_of*', are used to annotate the activities with domain ontology. '*Equivalent_name'* provides synonym of the name from terminology level using thesaurus. However, we can add other relationships such as '*is_as'*, '*step_of*', '*subClass_of*' and '*instance_of*' to relate the concept of Activity with concepts defined in domain ontology.

The relationships: '*has_Wf-Actor'*, '*has_Wf-Artifact'*, '*has_Wf-Input'*, '*has_Wf-Output'*, denote the relationships between the activity and other related concepts in CWO. The ontology concepts are denoted by URI (Uniform Resource Identifier) in the CPSAM.

- A role is an organizational concept, which may be a supervisor or a manger in an organization that interacts with an activity. The annotated role is represented as follows :

  ARi = (id, name, same_as, equivalent_name, member_of, subClass_of, instance_of).

- An actor is a person, agent or user that that participates in the process execution. The annotated role is represented as follows.

  ACi = (id, name, same_as, equivalent_name, member_of, subClass_of', instance_of).

- An artifact is an object (document, sheet styles, electronic form, etc.), which is manipulated by an activity or a person. It is described as follows :

  AFi = (id, name, same_as, equivalent_name, related-activity).

- The inputs and outputs are defined as parameters of an activity, which include data type. They are usually related to artifacts participating in the activity.

  INPi = (id, name, same_as, equivalent_name, data-type, related-artifact).

  OUPi = (id, name, same-as, equivalent_name data-type, related-artifact).

- A resource is a physic entity in an organization (tool, machine, etc.) that may be invoked by an activity during the process execution. it is described as follows :

  RSi = (id, name, same_as, equivalent_name, related-activity).

- The transition conditions are represented by expressions to constrain the inputs and outputs of the activities. They enable to specify the flux control, which is expressed with logical symbols like AND, OR, and XOR.

  ATi = (id, name, same_as, equivalent_name, related_activity).

7 Model annotations for semantic discovery purposes

In a context of inter-enterprise cooperation, the distributed process models should be accessible and reuse in specific projects. For enhancing the interoperability and especially for profiles or goals-oriented queries, we need the consensual representations to specify the semantics of profiles or goals of the models. Therefore, we complement the proposed semantic annotations with profiles and goals annotations.

7.1 7.1 Profiles annotation

A profile provides a general description of a process model, intended to be published and shared for

facilitating its discovery. Profiles can include both functional properties (inputs, outputs, preconditions, and post-conditions) and non functional properties (model name, text description, contact information, the location of the model, etc.).

Models profiles concepts are used to annotate intentional usage of process models. In our approach, we build a common profile ontology, which is used to annotate profiles concepts of models. We use the relationship 'related_Wf-Profile' to annotate the models with the common profile ontology. In this way, the common process semantic annotation model (CPSAM) will be extended with this type of annotation and will be formalized as follows:

CPSAM = (AV, AR, AC, AF, AE, AS, AT, RS, DO, PO).

Where PO is a subset of profile ontology concepts and an annotated activity AVi is described as follows:

AVi = (id, name, same_as, equivalent_name, has_Wf-Role, has_Wf-Actor, has_Wf-Artifact, has_Wf-Input, has_Wf-Output, has_Wf-Transition, has_Wf-Resource, kind_of, phase_of, related_Wf-Profile).

7.2 Goals annotation

A goal can be linked to a process model (or a process model fragment). The aim of goal annotation is to facilitate the goal-driven process discovery. The goal annotated models with the global (or common) goal ontology can be queried and reused in specific projects for achieving business objectives (or goals).

Goal annotation of process models is annotating process models with global goal ontology to specify the objectives of processes. Goal ontology is therefore, a set of concepts and relationships between the goals. However, few processes modeling languages and tools support the modeling of the goals as a part of processes modeling, e.g. EEML (Extended Enterprise Modeling Language) [26], which is implemented in Metis tool [27]. Thus, the goals can be modeled and linked to elements of process models. Nevertheless, the representation of the goals and relationships between the goals and the processes are not explicitly represented in many process models.

In the literature of processes models, several goal-oriented modeling approaches [28], [29], [30], [31] were proposed for defining the goal concept. KAOS (Knowledge Acquisition in autOmated Specification) [32] and i*/GRL (Goal-oriented Requirement Language) [33] are considered in the community of the artificial intelligence, and in particular in the requirements engineering as the most significant approaches allowing to model the objectives. The concepts defined in KAOS are: object, action, agent, goal, constraint. Contrary in i*/GRL, the concepts are the following: actor, role, position and goal.

To express goals, [28] defines verbs key words like those used by KAOS such as: achieve, avoid, maintain. In fact, the use of these verbs leads to a classification of the goals: hard goals and soft software [29], [35]. The hard goals relate to functional goals which are supported by the processes. A functional need defines a potential need that the system must satisfy. The soft goals are related to non functional needs concerning the additional qualities awaited such as performance, quality of service, cost and time.

In a perspective of a goal-oriented cooperation between companies, the creation of a global goal ontology is needed, which provides semantic representations of goals in a consensual way for different organizations. It enables us to build relationships between the local goals and the global goals. Thus, two types of relationships are created to indicate that activity can achieve goals: one relation of type 'achieves_local_but' between the activity and the local goal, and another relation of type 'achieves_global_but' between the activity and the global goal. That means obviously that the process meta-model must preliminary integrate the concept of local goal to facilitate the annotation. Thus, the common process semantic annotation model (CPSAM) will be enriched and extended with this type of annotation and will be formalized as follows:

CPSAM = (AV, AR, AC, AF, AE, AS, AT, RS, DO, PO, GO).

Where GO is a subset of global goal ontology concepts and an annotated activity AVi is described as follows:

AVi = (id, name, same_as, equivalent_name, has_Wf-Role, has_Wf-Actor, has_Wf-Artifact, has_Wf-Input, has_Wf-Output, has_Wf-Transition, has_Wf-Resource, kind_of, phase_of, related_Wf-Profile, achieves_global_but).

However, the goals ontologies belong to the class of the tasks ontologies, which describe tasks or activities for achieving the business objectives. In this context, the tasks ontologies are regarded as goals ontologies. For example, the tasks ontology SCOR contains concepts of goals [36], [20], which are formalized by hard goals and soft goals.

- **SCOR as example of global goal ontology**

The goal ontology in our Workflow domain is also from the SCOR. Usually the hard goals are derived from the level 3 process elements [36] and their inputs and outputs. The performance attributes defined in SCOR are General Soft Goal Category such as reliability, responsiveness, flexibility, cost, and assets. The domain specific soft goals are derived from the metrics of the performance attributes [37]. By analyzing SCOR ("S1 Source Stocked Product" model), several types of hard goals can be derived and extracted from the level 3

process elements [36]. We can quote some examples of goal-oriented activity (called 'hard goals') : *'Sourced Product are On Order'; 'Order is Placed', 'Order is Validate', 'Order is Consolidated'; 'Order is Processed', 'Available Inventory', 'sourced products are verified', 'End Items are Delivered'.* Also, there exist *goal-roles-oriented* such as *"Procurement Notification to Supplier' and 'Payment is Authorized to Supplier'.*

For example, two Workflow models are both about purchase order process domain. The activities for these models: *'Check Items'* and *'Verify Order', which are annotated as* a kind of the activity ontology '*Verify Product'*, will be both annotated with the domain goal ontology (i.e, SCOR) by a hard goal *'sourced products are verified'* and also by a soft goal '*decrease % defective supplied'* .

Additionally to these main annotation sets for semantic interoperability of Workflow models, namely, meta-model annotation, model annotation, profiles and goals annotation, we introduce another type of annotations.

8 Model aspects annotation

In order to enhance interoperability and to enable a common understanding of the models without any ambiguity; we enrich our semantic annotation framework with complementary annotations that are related to these models. According to [43], [44,] there are different aspects of business process: the informational, the transaction, the security, the quality of services aspect, etc. Which of them are applicable or relevant depends on the context and the purpose of the model. For our concern, we quote the basic aspects: the informational, the functional, the organizational, the behavioural or the resources aspect.

This type of annotation is useful and can then serve for the interoperating actors to know, for example:

1) In which the business process is executed, its organizational units and which role or a person of appropriate skills must perform one particular business activity?.

2) The order of the execution in which the corresponding activities must be executed.

3) What information involved in a business process and how artifacts are propagated among activities?.

4) Which resources are required (external applications, computer tool, etc.) for the execution of activities?.

In our work, the proposed annotations refer to several common ontologies. Each one is derived from a common meta-model corresponding to one aspect. Indeed, as argued in [38], the notion of meta-model is strongly related to the notion of ontology. Thus, these common meta-models are considered as common ontologies and implemented in OWL.

8.1 Informational annotation

The information about models is usually represented in the artifacts manipulating by the activities and containing data, the inputs and outputs as parameters of activities. To annotate the models information, we refer to a common informational ontology, which include the concept 'Wf-Artifact' and the concepts 'Wf-Input' and 'Wf-Output'. As we have argued before, this ontology is derived from a common informational meta-model, which is shown in figure 5.

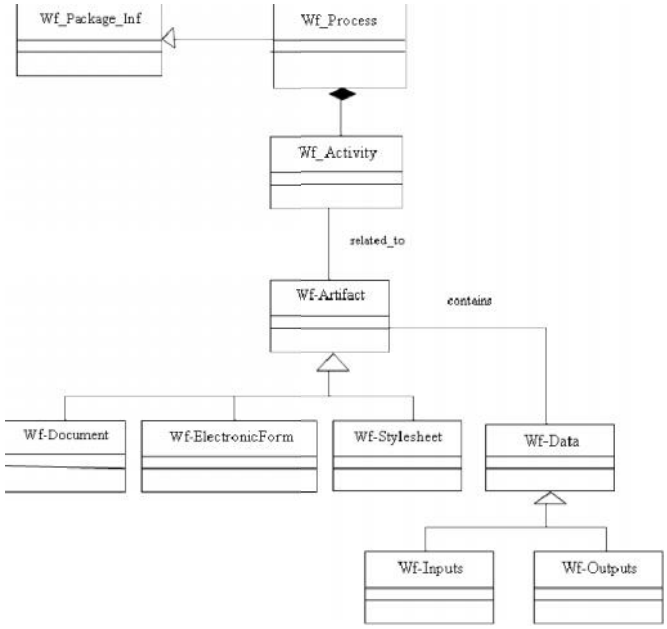


Figure 5: Common Informational Meta-model.

To annotate the activities with a common informational ontology, we use the relationship '*related_Wf-Artifact'*. Thus, an annotated artifact and an annotated activity AVi are described as follows:

Wf-Artifact = (id, name, equivalent_name, same_as, related_Wf-Input, related_Wf-Output).

AVi = (id, name, same_as, equivalent_name, kind_of, phase_of, related_Wf-Artifact).

8.2 Functional annotation

The purpose of the functional annotation is to make explicit the meaning of the functional aspect of a process model. This type of annotation is about the operations (so-called functions) of a process.

Back to the purchase order process domain, when annotating the purchase order process, the annotator refers to an UML activity diagram (figure 6) that contains all the activities of a process. For our case, these activities represent the following operations : `CreateOrder', `CheckOrder', `CancelOrder', `NotifyCustomer', `FillOrder', `ShipOrder', `CloseOrder', which enable to specify the functional aspect of a purchase order process.

To annotate the functional aspect, we refer to a common functional ontology, which is derived from a

common functional meta-model (figure 7). It includes the various functions associated to the activities of a process model. These functions usually correspond to operations carried out by a process. In our approach, a function correspond to an operation (or verb representing the operation) and an object (for instance, an order) contributing to the operation execution.

Therefore, to link the functions of activities (or operations) to a common functional ontology, we use of the relationship '*related_Wf-function'*. Thus, an annotated function and an annotated activity AVi are described as follows:

Wf-Function= (id, name, equivalent_name, same_as)

AVi = (id, name, same_as, equivalent_name, kind_of, phase_of, related_Wf-Function).

8.3 Behavioural annotation

The behavioural annotation concerns the dynamic aspect of a process. The purpose of such type annotation is, for example, to ensure that cooperating processes have the same behavior (then one can be used instead of the other). In our approach, this type of annotation is provided under the form of enumeration of various passed statecharts by the activities of the process and under form of flow control. To express the common behavior of process models, two common meta-models are proposed:

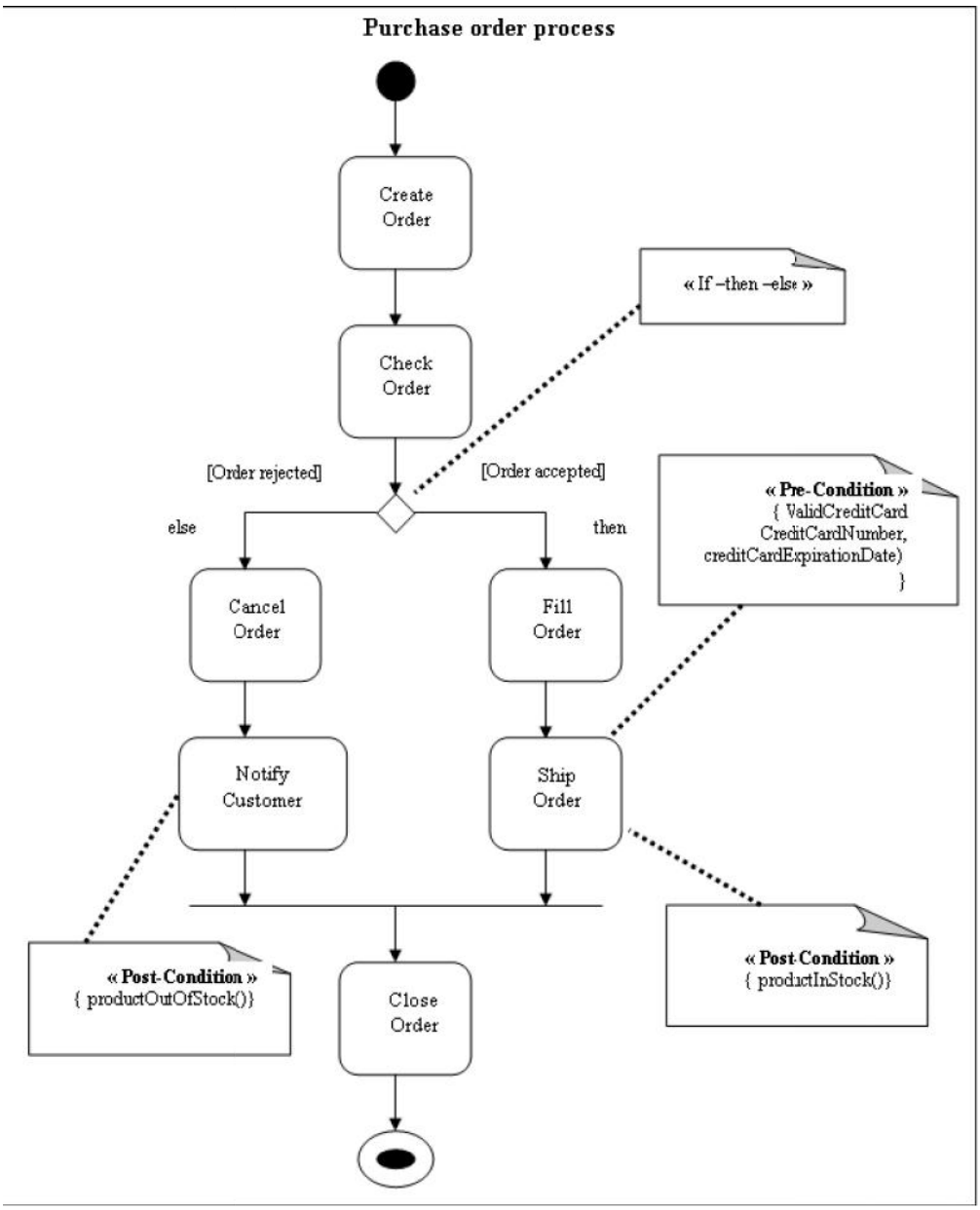


Figure 6: UML Activity Diagram of a Purchase Order Process.

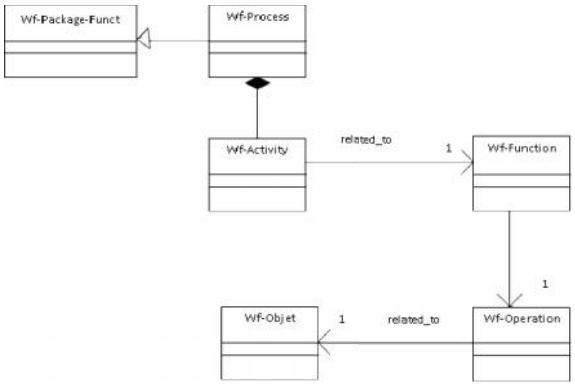


Figure 7: Common Functional Meta-model.

i) The first one (figure 8) serves to specify the flow control. It enables in particular, to define the flow control, which is usually expressed under form of sequence, parallel and choice.

ii) The second meta-model (figure 9) is inspired by the statecharts diagram of UML meta-model whose the basic concepts are: StateMachine, State, Transition, Vent, Guard and whose constraints are expressed and formalized by language OCL (Object Language Constraint) associated with UML.

To annotate the dynamic aspect of process models, we refer to both common behavioural ontologies, which are derived from common behavioural meta-models and presented above. The control flow represents the type of the ordering of activities, which is usually described by six operators AND Join, AND Split, XOR Join, XOR Split, OR Join, OR Split. It is described as follows:

CFi = (id, name, equivalent_name, related_Wf_Activity).

Three types of flow control may be distinguished under form of sequence, parallel and choice. For an activity *AVi*, we can describe these three flow control as follows:

$Sequence_i$ *= (id, name, equivalent_name, has_inActivity, has_outActivity).*

$Choice_i$ *= (id, name, equivalent_name, has_inActivity, has_outActivity, has_logicConnecteur).*

$Choice_i$ corresponds to the junction or the disjunction at inputs of the activities $(Join_i)$ and the element *`has_logicConnecteur'* of $Choice_i$ has the value `XOR' for *XOR* $Join_i$, the value `OR' for *'OR* $Join_i$*'* and the value `AND' for *AND* $Join_i$

$parallel_i$ *= (id, name, equivalent_name, has_inActivity, has_outActivity, has_logicConnector).*

$parallel_i$ corresponds to the junction or the disjunction at outputs of the activities ($Split_i$) and the element *`has_logicConnecteur'* of $parallel_i$ has the value `XOR' for *XOR* $Split_i$, the value `OR' for *OR* $Split_i$ and the value `AND' for *AND* $Split_i$.

For a certain activity *(AVi)*, the inputs (Inp_i) and the outputs (Oup_i) are defined as parameters of' activity, which include data type. They are usually associated with artifacts manipulated by the' activity. We can describe them as follows:

Inp_i *= (id, name, equivalent_name, data-type, related_Wf_artifact).*

Oup_i *= (id, name, equivalent_name, data-type, related_Wf_artifact).*

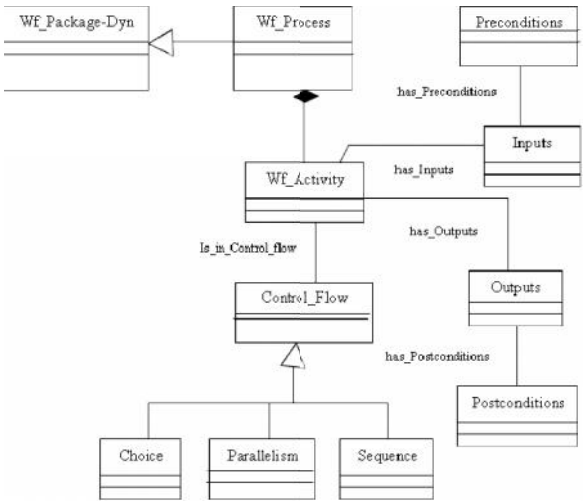


Figure 8: Common Behavioural Meta-model (flow control).

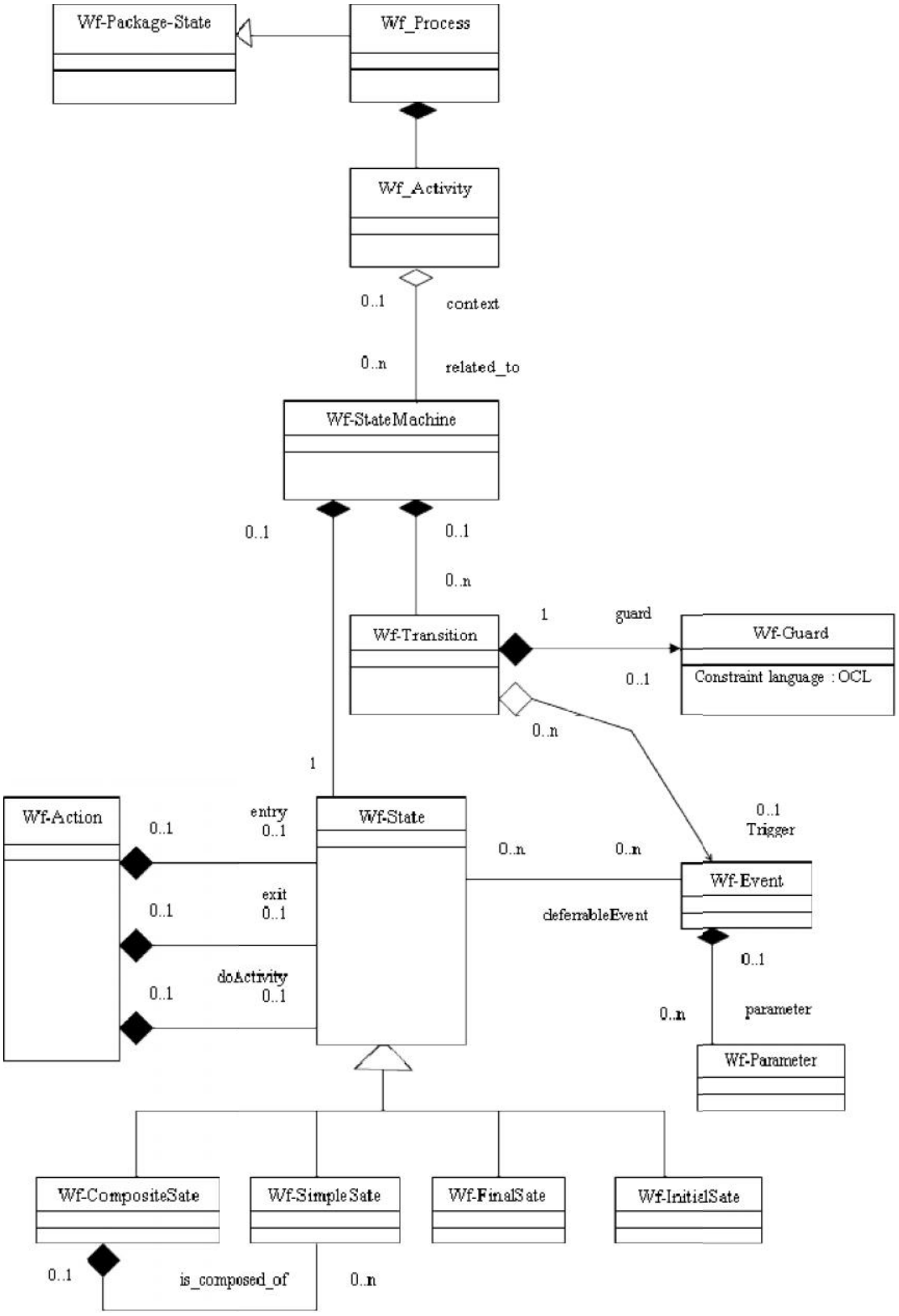


Figure 9: Common Behavioural Meta-model (Statecharts).

In the same way, the preconditions *(Pre_i)* and the post-conditions *($Post_t$)* of the common behavioural meta-model including the flow control can be described as follows :

Pre_i = (id, name equivalent_name, related Wf_input).

$Post_t$ = (id, name, equivalent_name, related Wf_output).

Also, the dynamic aspect of process models is expressed through the various states/transitions which we have expressed by the common meta-model of statecharts. Thus, an activity is annotated by referring to its state of transition via the relation " related_Wf-StateMachine". Therefore, an annotated activity *AVi* is described as follows:

AVi = (id, name, equivalent_name, same_as, kind_of, phase_of, related_Wf-StateMachine).

8.4 Organizational annotation

To annotate the organizational aspect of models, we refer to a common organizational ontology, which is derived from common organizational meta-model (figure 10) using the relationship '*related_Wf-OrganizationalUnit'*. This meta-model includes oganizationals concepts such as organizational unit (department, service, etc.), role, and actor. Thus, an annotated OrganizationalUnit and an annotated activity AVi are described as follows:

Wf-OrganizationalUnit= (id, name, equivalent_name, same_as, is_composed_of).

AVi = (id, name equivalent_name, same_as, kind-of, phase_of, related_Wf-OrganizationalUnit).

The relationship '*is_composed_of* ' refers to organizational concepts defined in common organizational ontology. The annotation of an organizational concept such role, for example, is described as follows:

Wf_Role = (id, name equivalent_name, same_as, member_of, instance_of)

8.5 Resources annotation

This type of annotation relates to the resources, which are invoked by an activity or a process for its execution. These resources can be material (editor, etc.), software (programs, external applications, etc.), human (well defined qualified persons) or temporal (time envisaged and carried out by the activity). The annotation of resources refers to a common resources ontology, which is derived from a common resources meta-model (figure 11).

We use the relationship '*related_Wf-Resource'* for linking the resources of activities to this ontology". Thus, an annotated resource and annotated activity AVi are, therefore, described as follows:

Wf-Resource = (id, name, equivalent_name, same_as).

AVi = (id, name, equivalent_name, same_as, kind_of, phase_of, related_Wf-Resource).

To annotate an activity with reference to several common ontologies, we extend our CPSAM by adding a set of common concepts related to models aspects. Thus, the CPSAM will be formalized as follows:

CPSAM = (AV, AR, AC, AF, AE, AS, AT, RS, DO, PO, GO, IO, FO, BO1, BO2, OO, RO).

Where DO is a subset of Domain Ontology concepts, PO is a subset of Profiles Ontology concepts, GO is a subset of global Goal Ontology concepts, IO is a subset of Informational Ontology concepts, FO is a subset of Functional Ontology concepts, BO1 is a subset of Behavioural Ontology1 (related to flow control) concepts, BO2 is a subset of Behavioural Ontology2 (related to statesharts) concepts, OO is a subset of Organizational Ontology concepts and RO is a subset of Resources Ontology concepts.

Now, we can say that for annotating an activity *(AVi)* of a process model, we use different relationships to link this activity to common ontologies (i.e. relating the activity with concepts defined in domain ontology, goal ontology, informational ontology, etc.).

Therefore, an annotated activity *AVi* is described as follows:

AVi = (id, name, same_as, equivalent_name, has_Wf-Role, has_Wf_Actor, has_Wf-Artifact, has_Wf_Inputs, has_Wf_Outputs, has_Wf_Preconditions, has_Wf_Postconditions, is_in_Control_Flow, has_Wf-State, has_Wf-Resource, kind_of, phase_of, related-Wf_Profile, achieves_global_but).

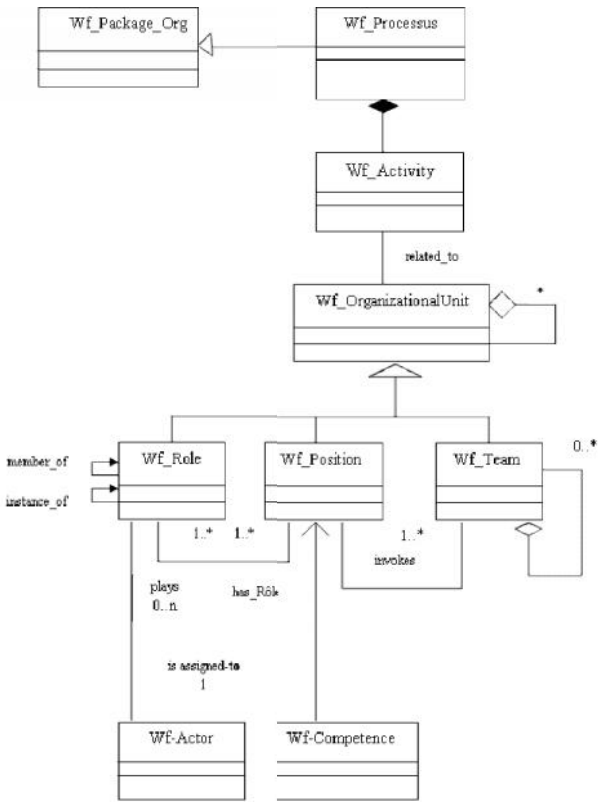


Figure 10: Common Organizational Meta-model.

The relationships: (*'has_Wf-Role' et 'has_Wf_Actor'*) refer to an organizational ontology. The relationships: *('has_Wf-Artifact', 'has_Wf_Inputs', 'has_Wf_Outputs',)* refer to an informational ontology. The relationships: *('has_Wf_Preconditiuons', 'has_Wf_Postconditions', 'is_in_Control_Flow', 'has_Wf-State*') refer to a behavioural ontology1 (flow control) and a behavioural ontology2 (statescharts) and the relationship: *'has_Wf-Resource'* refers to resources ontology.

As for the relationships: (*'kind_of'*,'phase_of'), they enable us to annotate the activity with the SCOR domain ontology and the relationships :(*'related-Wf_Profile' and 'achieves_global_but'*), they are used to link the activity to the profile ontology and the goal ontology, which is also SCOR in our case.

Based on the above formalization and from a technical view point, these annotations are codified in a XML language with namespace CWO (Common Workflow Ontology), and they are sent by the actor *Act1* to the actor *Act2* for interpretation of the activity.

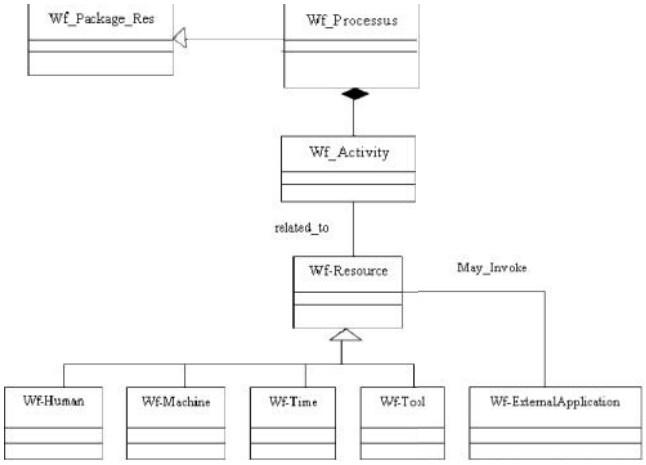


Figure 11: Common Resources Meta-model.

9 Conclusion and future work

The interoperability problem is very complex. It is still more complex in the Workflow domain (or process models), which overlaps several fields of application such as the software engineering, the enterprise modeling, the Workflows systems and the Web services composition. This complexity is mainly due to several aspects, which are by nature related to any Workflow process. Among these aspects, we quote those which are considered basic aspects and concern: the informational, the functional, the behavioural, the organizational or the resources type.

However, the problem of the semantic interoperability of distributed process models can occur in a homogeneous and in a heterogeneous context. To manage this problem on the conceptual level, we have proposed an agnostic approach based on the use of the semantic annotations techniques.

In a homogeneous context, we have used two types of annotations : *1) structural annotations :* by using only the notation UML, and referring the described concepts in either same ontologies where there is really a consensus between all the partners for all shared concepts, or in ontologies of the partners if there are not completely identical. *2) Lexical annotations:* by referring to a lexical database such as WordNet, or a WfMC's (Workflow Management Coalition) glossary that contains all the basic vocabulary used in the field of Workflow.

However, to manage the semantic heterogeneity of models within a heterogeneous context, we have proposed a semantic annotation framework allowing the interoperating actors to semantically annotate their Workflow models at different levels: *1)* meta-models *2)* models content *3)* profiles *4)* goals *5)* models aspects such as the organizational or the informational type.

Among the advantages that our approach offers, we note some of them. First, by externalizing the semantic domain models, we take an agnostic approach to ontology representation languages, as well as of any Web technology. This allows the cooperating actors to annotate their models (or models fragments) according to their preferred ontology representation language. Secondly, by keeping the semantic annotation mechanism separate from the representation of the semantic descriptions, the approach offers flexibility and openness to the developers' community to select their favorite semantic annotation language such as WSDL-S or OWL-S. In addition, this approach enables the traceability of models thanks to their annotations by the successive transformations that may be applied to some initial models.

Obviously, much work in practice for testing the feasibility of the approach and its applicability through applications in business process models integrations in the real world.

For our concern, we have just presented in this paper a theoretical solution for a semantics interoperability problem of process models in a homogeneous and in a heterogeneous context. Certainly, to disclose the technical possibility of the approach, we envisage in future paper to provide a selected case study in an industrial enterprise.

Therefore, in our future investigations, we are planning to continue our work. First, we implement the different common ontologies in OWL. Second, we create a semantic annotation tool according to our framework, which is currently under development. Then, we apply our semantic annotation procedure as described in section 3.2.

Finally, to support the model annotation, we think that a platform should be developed including, in an integrated manner, facilities or services for the ontology management, the annotation management, etc. (like querying, match-making or browsing an ontology), which where be coupled with annotations services.

10 References

[1] S. Hamri, N. Boudjlida and M. Boufaida (2007). "An approach for building an OWL Ontology for Workflow Interoperability". In

Proceedings of the 3rd International Conference on Interoperability of Enterprise Software and Applications, Madera, Portugal, 26-30 March 2007, p. 357—364. R.J. Gonçalves, J.P. Müller, K. Mertins and M. Zelm eds, Springer-Verlag, ISBN: 978-1-84628-857-9.

[2] S.Hamri (2009). "Bridging MDA and OWL for Workflow Interoperability", In International Review on Computers and Software (IRECOS), ISSN 1828-6003. Vol.4, No. 3, pp 374-381.

[3] "ebXML (2001), "Business Process Specification Schema Version 1.01", ebXML Business Process Project, http://www.ebXML.org/specs/ebBPSS.pdf.

[4] D.A Bensaber and M. Malki (2012), "Model Driven Approach for Specifying WSMO Ontology", In Proceedings of the 4th Int. conference on Web and Information Technologies, pp 203-213, Algeria,

[5] F.Leyman (2001). "Web Services Flow Language ", IBM Software Group specification, http://www306.ibm.com/software/solutions /webservices/pdf/WSFL.pdf.

[6] OMG (2003), Model Driven Architecture, MDA Guide version 1.0.1, http://www.omg.org /mda.

[7] BPEL4WS (2005). Business Process Execution Language for Web Services version 1.1". http://www.128.ibm.com/developerworks/library/specification /ws-bpel/,

[8] Workflow Management Coalition (2001). Workflow Process Definition Interface, XML Process Definition Language, Document Number TC-1025.

[9] WfMC (2000). Workflow standard-interoperability, wf-xml binding. Wfmc-tc-1023, v. 1.0.

[10] OWL-S (2004): OWL Services Coalition OWL-S: Semantic Markup for Web Services, http://www.daml.org/services/owl-s/1.1.

[11] WSMO(2005), http://www.wsmo.org/TR/d16/ d16.1/ v0.21.

[12] WSDL-S (2005), http://lsdis.cs.uga.edu/projects/ WSDL-S/wsdl-s.pdf.

[13] J. César et al (2006). " Modeling Semantic Web Services: A Case Study", ICWE'06, ACM Palo Alto, USA, pp.32-39.

[14] J.T.E. Timm, G. C. Gannod (2008). "Grounding and Execution of OWL-S Based Semantic Web Services", IEEE International Conference on Services Computing, Vol. 2, pp.588-592.

[15] J.T.E.Timm and G.C. Gannod (2005). 'A model-driven Approach for Specifying Semantic Web Services', Proceedings of the 3rd International Conference on Web Services, pp. 313-320,

[16] G.C. Gannod, et al (2006). ''Facilitating the Specification of Semantic Web Services Using Model-Driven Development '' in International Journal of Web Services, Vol 3, No 3, pp.61-81.

[17] Y. Lin (2008). " Semantic Annotation for Process Models: Facilitating Process Knowledge Management via Semantic Interoperability". PhD thesis, Norwegian University of Science and Technology.

[18] Y. Lin, H. Ding (2005)." Ontology-based Semantic Web Annotation for Semantic Interoperability of Process Models". In Proc. of the Int'l Conf. on Computational Intelligence for Modeling, Control and Automation, Vienna, Austria.

[19] Y. Lin et al (2006). "Semantic Annotation Framework to Manage Semantic Heterogeneity of Process Models". In: Dubois, E., Pohl, K. (eds.) CAiSE. LNCS, vol. 4001, pp. 433–446.

[20] Y. Lin and A.Sølvberg (2007)." Goal annotation of process models for semantic enrichment of process knowledge". In Proc. of the 19th International Conference on Advanced Information Systems Engineering (CAiSE 2007), pp 355–369, Norway. LNCS 4495.

[21] G.Miller et al (1993), "Introduction to WordNet: An on-line lexical database", http://wordnetweb. princeton.edu/perl/webwn.

[22] http://wikipédia.org.

[23] Workflow Management Coalition (1996), Terminology and Glossary, WfMC Document Number TC-1011, issue 2.0.

[24] WfMC, http://www.wfmc.org (1996).

[25] Protégé, http://protégé.stanford.edu/plugin.

[26] J.Krogstie,, D.H Jørgensen (2004)." Interactive Models for Supporting Networked Organisations, In Proc. 16th Int. Conf. on Advanced Information Systems Engineering, LNCS 3084, Springer-Verlag 550-562.

[27] http://www.troux.com (2006).

[28] A.V. Lamsweerde, (2001) "Goal-oriented requirements engineering: a guided tour". In 5th International Conference on Requirements Engineering, pages 249–262.

[29] N.Mellal (2007). "Réalisation de l'interopérabilité sémantique des systèmes, basée sur les ontologies et les flux d'information", Thèse de Doctorat, Université de Savoie (France).

[30] Yu, E. J. Mylopoulos (1998)."Why goal-oriented requirements engineering", In: Proc. of the 4th of International Workshop on Requirements Engineering: Foundations of Software Quality http://www.cs.toronto.edu/pub/eric/REFSQ98.html.

[31] M.Lin, H.Guo, J.Yin (2005)." Goal description language for semantic web service automatic composition", In: Proc. of IEEE/IPSJ International Symposium on Applications and the Internet (SAINT 2005), Trento, Italy. pp. 190–196.

[32] R.Darimont et al. (1997). "Grail/kaos: An environment for goal-driven requirements analysis, integration and layout".

[33] i*: An agent-oriented modelling framework, (2008). http://www.cs.toronto.edu/km/istar/

[34] C. Diamantini and N. Boudjlida (2006). "About Semantic Enrichment of Strategic Models as Part of Enterprise Models. " In J. Eder and S. Dustdar Eds, Proceedings of the 2nd Interntional Workshop on Enterprise and Networked Enterprises Interoperability, in conjunction with the 4th International Conference on Business Process

Management. Pages 348--359. LNCS# 4103. Vienna, Austria.

[35] J. Mylopoulos, L. E.Yu. Chung (1999). "From object-oriented to goal-oriented requirements analysis. Commun", ACM 42, 31–37

[36] SCOR (2006): SCOR model, version 7, the Supply Chain Council. http://www.supply-chain.org/page.ww?section=SCOR+Model&name=SCOR+Model.

[37] P.Soffer, Y.Wand (2005). "On the notion of soft-goals in business process modeling", Business Process Management Journal 11, pp 663–679.

[38] J. Bézivin (1998). "Who‘s Afraid of Ontologies”, OOPSLA’98 Workshop, Model Engineering, Methods and tools, Integration with CDIF’’, Vancouver, October, http://www.metamodel.com/oopsla_cdif-workshop/bezivin1/

[39] N. Boudjlida and H. Panetto (2007). "Enterprise Semantic Modelling for Interoperability", In Proceedings of the 12th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2007, September 25-28, Patras, Greece.

[40] M. Elias, et al (2010). “A Business Process Metadata Model for a Process Model Repository. In: Bider, I., Halpin, T., Krogstie, J., Nurcan, S., Proper, E., Schmidt, R., Ukor, R. (eds.), Lecture Notes in Business Information Processing, vol. 50, pp. 287–300, Springer.

[41] M.Elias, P.Johannesson (2012). “An Empirical Assessment of the Effect of Context-Based Semantic Annotation on Process Model Discovery’’.CAiSE Workshops 2012: 366-382, Gdańsk, Poland, June 25-26, ISBN 978-3-642-31068-3, Springer,.

[42] M.Elias, P.Johannesson (2012). “A survey of process model reuse repositories’’. In: Dua, S., Gangopadhyay, A., Thulasiraman, P., Straccia, U., Shepherd, M., Stein, B. (eds.) ICISTM, Communications in Computer and Information Science, vol. 285, pp. 64–76, Springer.

[43] B. Axenath, E. Kindler and V. Rubin (2005). “The Aspects of Business Processes: An Open and Formalism Independent Ontology”, Technical report, Department of the Unversity of Paderborm, Germany, http://www.upb.de/cs/kindler/publications/copies/AKR05.pdf

[44] J.Gray and A.Reuter (1993). "Transaction Processing : Concepts and Techniques ".

[45] M. Dimitrov, Alex Simov, Sebastian Stein, and Mihail Konstantinov (2007). "A BPMO Based Semantic Business Process Modelling Environment". In Workshop on Semantic Business Process and Product Lifecycle Management, volume 251 of CEUR Workshop Proceedings. CEUR-WS.org,

[46] M. Born, , et al (2009). "Supporting Execution-level Business Process Modeling with Semantic Technologies. " In Proc. of the 14th DASFAA Conference. LNCS 5463, Springer.

[47] C. Di Francescomarino, Chiara Ghidini, Marco Rospocher, Luciano Serafini, and Paolo Tonella (2008). "Reasoning on semantically annotated processes". In Proceedings of the 6th International Conference on Service-Oriented Computing, vol 5364 of Lecture Notes in Computer Science, pages 132–146.

[48] D. Calvanese, , et a (2012). "Ontology-Based Governance of Data-Aware Processes". In Proc. of the 6th Int. Conf. on Web Reasoning and Rule Systems, LNCS 7497, Springer.

[49] F. Smith, Proietti, M. (2013). "Rule-based Behavioral Reasoning on Semantic Business Processes". In Proc. of the 5th Int. Conf. on Agents and Artificial Intelligence, SciTePress.

Incremental Hierarchical Fuzzy Model Generated from Multilevel Fuzzy Support Vector Regression Network

Ling Wang, DongMei Fu and LuLu Wu
School of Automation & Electrical Engineering, University of Science and Technology, Beijing, 100083, China
Key Laboratory of Advanced Control of Iron and Steel Process (Ministry of Education), Beijing, 100083, China

Keywords: FCM clustering, fuzzy association rules, incremental hierarchical fuzzy model, multilevel fuzzy support vector regression network

Received: March 23, 2014

Fuzzy rule-based systems are nowadays one of the most successful applications of fuzzy logic, but in complex applications with a large set of variables, the number of rules increases exponentially and the obtained fuzzy system is scarcely interpretable. Hierarchical fuzzy systems are one of the alternatives presented in the literature to overcome this problem. This paper presents a multilevel fuzzy support vector regression network (MFSVRN) model that learns incremental hierarchical structure based on the Takagi-Sugeno-Kang(TSK) fuzzy system with the aim of coping with the curse of dimensionality and generalization ability. From the input–output data pairs, the TS-type rules and its parameters are learned by a combination of fuzzy clustering and linear SVR in this paper. In addition, an efficient input variable selection method of the incremental multilevel network is proposed based on the FCM clustering and fuzzy association rules. To achieve high generalization ability, the consequence parameters of a rule are learned through linear SVR with a new TS-kernel. This paper demonstrates the capabilities of MFSVRN model by conducting simulations in function approximations and a chaotic time-series prediction. This paper also compares simulation results from the single-level counterparts-FSVRN and Jang's ANFIS model.

Povzetek:Predstavljen je hierarhični mehek model, zgrajen iz večnivojske mehke regresijske mreže.

1 Introduction

As is widely known, fuzzy rule-based systems have been proposed and successfully applied to solving problems such as classification, identification, control, etc. At present, one of the important issues in fuzzy rule-based systems is how to reduce the total number of involved rules and their corresponding computation requirements. In a single fuzzy system, in order to maintain a specified accuracy, the number of rules grows exponentially with the number of input variables or input fuzzy sets, but the interpretability of the fuzzy system is lost. Hence, to deal with the "curse of dimensionality" and the rule-explosion problem, hierarchical fuzzy system (HFS) was proposed [1] and have attracted much attention in recent years. Most hierarchical fuzzy systems [2-9] consist of a number of low-dimensional fuzzy systems in a hierarchical form, the number of rules can be reduced to as low as being a linear function of the number of input variables. A HFS is made up of a set of fuzzy subsystems or modules. These modules are linked such a way that the output of a module is the input of other ones. Fig.1 depicts some possible hierarchical models for 4 input variables and 2 hierarchical levels or 3 hierarchical levels. The incremental structure is shown in Fig.1(a), in which the inputs of a module is the output of the previous ones, along with external variables[2-6]. The aggregated structure is shown in Fig.1(b), in which the first level is made up of a set of modules receiving the input variables, and each variable is used as an input only in a single module. The outputs of the modules in the first level are the inputs of the modules which constitute the next level, and so on [7-9].

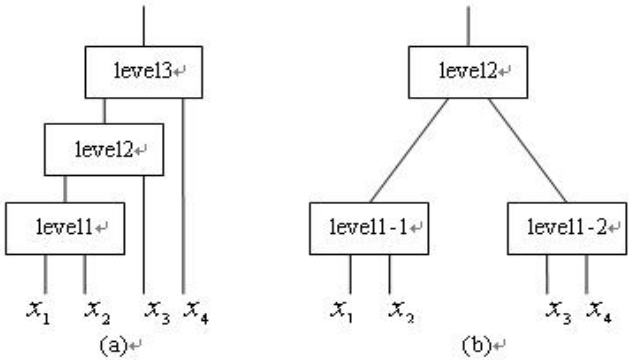


Figure1: Example of HFS Structure.

Our research focused on the incremental hierarchical structure. In this hierarchy, it is important to determine its input variables and their interactions in different levels. In general, by analyzing the relative importance of input variables, the most important input variables are assigned to the lowest level and the least important input variables are assigned to the highest level [12-13]. In order to assign the correlated or coupled input variables to the same level in hierarchical fuzzy system, Chung and Duan [13] introduced a correlation matrix to

determine the correlated or coupled input variables. In[22], a multi-objective genetic algorithm (MOGA) is adopted to determine the input variables of each level module and the hierarchical architecture. In [6], an evolutionary algorithm (EA) is investigated to select the input variables and the topology in HFS. But GA and EA are all very time-consuming searching process.

In the hierarchical fuzzy system design, each single-level module serves as individual reasoning level. As a fuzzy reasoning mechanism, Takagi-Sugeno-Kang (TSK) type is most widely used. The well-known Jang's ANFIS[21] model based on layered network architecture was firstly proposed. In [10] and [11], the neural-fuzzy network is also used to realize TSK fuzzy reasoning. Usually, the consequent in Takagi-Sugeno fuzzy model is a crisp function of antecedent variables, and the recursive least squares (RLS) is used to determine the parameters of nonlinear consequents. However, this further RLS tuning is based on empirical risk minimization and lack of the high generalization ability. The idea of solving this parameter estimation problem by incorporating SVR (support vector regression) into TSK model has recently attracted a lot of attention [23-24]. In [23], a SVR-based FNN was proposed that the fuzzy if–then rules are generated based on the extracted SVs. Since the number of SVs in an SVR is usually very large, the model size of a designed FS is equally large. In [24], fuzzy c -means (FCM) is used to generate fuzzy rules and their antecedent parameters, the consequent part parameters are obtained by SVR learning. In contrast to fuzzy neural network, the use of SVR for TSK learning has a smaller number of parameters and the use of kernel transformation retains the SVR's good generalization ability.

This paper proposes a multilevel fuzzy support vector regression network (MFSVRN) model to design a fuzzy system to solve the dimensionality problem and generalization performance. In order to achieve efficient input variables in each level, we adopt the FCM clustering algorithm [15] and fuzzy association rules mining method [19] to construct a MLFSVRN model with incremental architecture. In addition, based on neuro-fuzzy systems, a novel TSK type fuzzy reasoning system with support vector regression learning mechanism called fuzzy support vector regression network (FSVRN) is applied to the modules of hierarchical fuzzy system. To improve the prediction performance of hierarchical fuzzy system, the consequence parameters of a rule are learned through linear SVR with a new TS-kernel. In the view of single-level FSVRN, the overall network-Multilevel fuzzy support vector regression network (MLFSVRN) is formulated.

The paper is organized as follows: Section 2 briefly describes the data mining techniques used in this paper. Section 3 presents our research methodology including input selection algorithms and construction of the multilevel support vector regression network models. Section 4 shows the experimental results. Finally, Section 5 concludes the paper with our final remarks.

2 Related knowledge

2.1 Fuzzy C-Means (FCM) clustering

Fuzzy c-Means (FCM) clustering algorithm [15-16] is an iterative optimization algorithm that minimizes the cost function (1) subjecting to $u_{ij} \in [0,1]$;

$$J(U,v) = \sum_{i=1}^{n}\sum_{j=1}^{c} u_{ij}^{m} \left\| \vec{x}_i - \vec{v}_j \right\|^2 \tag{1}$$

Here, u_{ij} is the degree of membership of $\vec{x}_i$ in the cluster j, which is defined as (2)

$$u_{ij} = \frac{(1/\left\| \vec{x}_i - \vec{v}_j \right\|^2)^{1/m-1}}{\sum_{k=1}^{c}(1/\left\| \vec{x}_i - \vec{v}_k \right\|^2)^{1/m-1}} \tag{2}$$

where $\vec{v}_j$ is the center of the j th cluster,

$$\vec{v}_j = \frac{1}{\sum_{i=1}^{n}(u_{ij})^m} \sum_{i=1}^{n}(u_{ij})^m \cdot \vec{x}_i \tag{3}$$

$\vec{x}_i$ is the i th data sample, n is the number of data points; $m \in (1,\infty)$ is a weighting constant.

The optimal clusters are produced by minimizing the objective function.

2.2 Fuzzy association rules

For the numerical data, fuzzy association rules [19] are easily understood by humans because of the fuzzy termsets. In order to mine fuzzy association rules, we apply FCM clustering to transform each of the numeric variables into fuzzy sets (fuzzy partition) with its membership function u, and then these fuzzy partitions are used to generate fuzzy rules. Meanwhile, the centre of each fuzzy set and the maximum and minimum value for each partition of the input data points are determined by FCM. By finding the centre of each partition, we can label it very easily according to the data point at which the core occurs. The labelling of each partition is very important as it helps a lot in the -eventual generation of fuzzy association rules.

Given a set of records, n is the record number, two items $X = \{x_1, x_2, \ldots, x_p\}$ and $Y = \{y_1, y_2, \ldots, y_q\}$, p is the length of the itemset X, q is the length of the itemset Y. The fuzzy set membership value of variable x_j in the i th record is denoted $u(x_{ij})$. The Apriror approach is used for extracting fuzzy itemsets from a fuzzy data set based on interesting measures(support and confidence) and able to generate fuzzy association rules. A fuzzy association rule is an implication of the form $(X \text{ is } A) \Rightarrow (Y \text{ is } B)$. A and B are fuzzy sets that characterize X and Y respectively. The measures of support, confidence and correlation have been fuzzified for the purpose of fuzzy association rules. The fuzzy support of X is defined as follows.

$$f_{support}(X)=\frac{1}{n}\sum_{i=1}^{n}\prod_{j=1}^{p}u(x_{ij}) \quad (4)$$

Where $X=\left\{x_i \middle| x_i=\left(x_{ij}\middle| j=1,\cdots,p\right), i=1,\cdots,n\right\}$ .The fuzzy support of $X\Rightarrow Y$ is defined as follows.

$$f_{support}(X\Rightarrow Y)=\frac{1}{n}\sum_{i=1}^{n}\left(\prod_{j=1}^{p}u(x_{ij})\bigcap\prod_{j=1}^{q}u(x_{ij})\right) \quad (5)$$

The fuzzy confidence of $X\Rightarrow Y$ is defined as follows.

$$f_{confidence}(X\Rightarrow Y)=\frac{f_{support}(X\Rightarrow Y)}{f_{support}(X)} \quad (6)$$

The fuzzy correlation measure of the association rule $X\Rightarrow Y$ can be measured by computing

$$Correlation(X,Y)=\frac{P(X\bigcup Y)}{P(X)P(Y)}=\frac{f_{conference}(X\Rightarrow Y)}{f_{support}(Y)} \quad (7)$$

In order to mine fuzzy association rules, the definitions of fuzzy support and fuzzy confidence are used in Fuzzy Apriori instead of their crisp counterparts used in Apriori.

3 Multilevel fuzzy SVR network (MLFSVRN) modelling of incremental type

In order to determine a multilevel fuzzy SVR network model, we must determine the model structure and initial parameters. Based on an incremental type structure like the one shown in Fig.1(a), it is quite possible to consider some influential variables to the first level, the less influential ones to the next level, and so on. To do so, we must find a set of candidate variables, which play a significant role in determining the output.

3.1 Variable selection based on FCM clustering and fuzzy association rules

In this paper, fuzzy association rules are used to select more representative variables from the original ones to improve the later prediction performance. The different fuzzy confidence and fuzzy support values between input variables and output variables are examined by using fuzzy association rules. In this paper, cross-validation with best parameter grid search is adopted to obtain the best rules. we only report the best rules that are based on the fuzzy confidence value = 1 and the fuzzy support value = 0.03.

These rules are further ranked as their importance by

$$Importance(X,Y)=\log\left(\frac{f_{confidence}(Y\Rightarrow X)}{f_{support}(X)}\right)=\log\left(\frac{p(X/Y)}{P(X)}\right) \quad (8)$$

In particular, according to (8), the rule that has the importance value less than 0.8 is excluded. As a result, the extracted rules that relate to the important variables will be obtained. In order to determine the most influential input variables, the influence degree of variables are calculated. The term $ID(x_i)$ is used to represent the influence degree (ID) of x_i ,i.e., $ID(x_i)=\frac{important(x_i)+correlation(x_i)}{SUM_{ID}}$. Here, the term $important(x_i)$ is used to represent the best result of the degree of importance of the fuzzy association rules with x_i , which is done by (8), In a similar way, the term $correlation(x_i)$ is used to represent the best result of the degree of correlation of the fuzzy association rules with x_i , which is done by (7). For all extracted variables from fuzzy association rules, these two items are added up and described as

$$SUM_{ID}=\sum_{i=1}^{n}(important(x_i)+correlation(x_i)).$$

3.2 Constructing a MLFSVRN model with incremental architecture

Based upon the analysis method just described in Section 3.1, the influential input variables can be obtained and consequently a MLFSVRN model with incremental architecture can be constructed as shown in Fig.2. The construction algorithm of MLFSVRN Model with Incremental Architecture can be summarized as follows.

Step 1- Initialization:

The number of levels is h . This identified model is called model h and its output is denoted by $y^{(h)}$. All n input variables are put in a set S . Let $SUM_{ID}=\sum_{x_i\in S}\left(important(x_i)+correlation(x_i)\right)$. A threshold value T_{inc} is set to control the model structure(the number of levels). A large T_{inc} will set the combination of the representative input variables rigorously and hence generate complicated networks while a small value will set the combination of the representative input variables loosely and generate the networks with few sublevels. Such arrangement is used to make the first level reserve enough system information and let the first level contain at least two input variables.

Step 2- Determination of Level 1:

1) Choose n_1 most influential inputs as the input variables of the first level and write them as $x_i^{(1)}$'s. In order to make the first level contain enough system information, the value of n_1 is determined by

$$\sum_{i=1}^{n_1}ID(x_i^{(1)})=\sum_{i=1}^{n_1}\frac{important(x_i^{(1)})+correlation(x_i^{(1)})}{SUM_{ID}}<T_{inc} \quad \text{and} \quad n_1\geq 2 \quad (9)$$

According to (9), the first level of the model architecture contain at least two input variables. Here, the term $important(x_i^{(1)})$ is used to represent the best result of the degree of importance of the fuzzy association rules with $x_i^{(1)}$, the term $correlation(x_i^{(1)})$ is used to represent

the best result of the degree of correlation of the fuzzy association rules with $x_i^{(1)}$.

2) Set the level index $h=2$. Remove these n_1 input variables from S.

Step 3— Recalculation:

Recalculating SUM_{ID} and the influence degree values of the variables left in S, i.e.,

$$ID(x_i)=\frac{important(x_i)+correlation(x_i)}{SUM_{ID}}\ \forall x_i\in S \quad (10)$$

Step 4—Determination of level h:

Choose n_h most influential input variables, i.e., $x_i^{(h)}$'s, from S and assign them to level h. The number n_h is determined by $\sum_{i=1}^{n_h} ID(x_i^{(h)})>T_{inc}$ and $n_h\geq 2$.

Step 5—Termination:

Remove these n_h input variables from S. If S is empty, the algorithm terminates, otherwise go back to Step 3, let $h=h+1$.

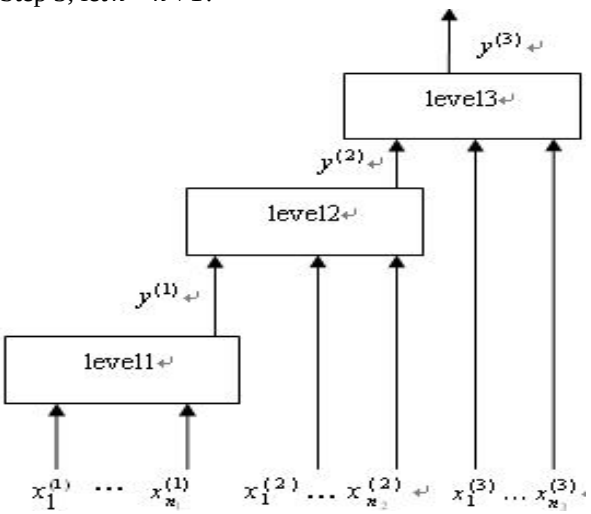


Figure 2: The most parsimonious incremental architecture for a three-level-input system.

3.3 TSK type MLFSVRN model structure and learning

3.3.1 TSK fuzzy inference algorithm

In this section, a TSK fuzzy inference algorithm in its each single-level module of MLFSVRN model is presented. Consider a TSK fuzzy system, the j th fuzzy rule in level h will have the form

$$\text{Rule}_j^{(h)}: \quad \text{IF} \quad \left(x_1^{(h)}\text{is}A_{1,j}^{(h)},\cdots,x_{n_h}^{(h)}\text{is}A_{n_h,j}^{(h)}\right) \text{ and } \left(y^{(h-1)}\text{is}B_j^{(h-1)}\right)$$

$$\text{THEN} \quad y^{(h)}=a_{0,j}^{(h)}+\sum_{i=1}^{n_h+1}a_{i,j}^{(h)}z_i^{(h)}. \quad (11)$$

Where

n_h is the total number of input variables (ordinary system inputs and input from previous level).

$x_i^{(h)}(i=1,\cdots,n_h)$ is input variables being determined by the proposed method in Section 3 and assigned to this level;

$y^{(h)}(h\leq H)$ is intermediate variables which represent the output from level h as well as the input to level $h+1$, H is the total number of levels;

$y^{(H)}$ is output variable;

$z_1^{(h)}=x_1^{(h)},\ z_2^{(h)}=x_2^{(h)},\ldots,z_{n_h}^{(h)}=x_{n_h}^{(h)},\ z_{n_h+1}^{(h)}=y^{(h-1)}$;

$a_{0,j}^{(h)}, a_{i,j}^{(h)}$ are crisp consequent part parameters;

Fuzzy set $A_{i,j}^{(h)}$ is employed with the following Gaussian-type membership function:

$$M_{ij}^{(h)}=\exp\left\{-\left(\frac{(z_i^{(h)}-m_{ij}^{(h)})^2}{\pm_j^2}\right)\right\} \quad (12)$$

Here, $m_{ij}^{(h)}$ and $\pm_j$ correspond to the center and width of the fuzzy set respectively, which are determined by FCM clustering.

The firing strength $\ll_j^{(h)}(\vec{z})$ of rule j is calculated by

$$\ll_j^{(h)}(\vec{z})=\prod_{i=1}^{n_h+1}M_{ij}^{(h)}=\exp\left\{-\sum_{i=1}^{n_h+1}\left[\frac{(z_i^{(h)}-m_{ij}^{(h)})^2}{\pm_j^2}\right]\right\}$$
$$=\exp\left\{-\frac{\left\|\vec{z}^{(h)}-\vec{m}_j^{(h)}\right\|^2}{\pm_j^2}\right\} \quad (13)$$

Where $\vec{z}^{(h)}=[x_1^{(h)},x_2^{(h)},\cdots,x_{n_h}^{(h)},y^{(h-1)}]$.

If the single-level fuzzy system contains r rules, then according to the simple weighted sum method [20], the output would be

$$y(\vec{x})=\sum_{j=1}^{r}\left[\ll_j^{(h)}(\vec{z})\cdot\left(a_{0,j}^{(h)}+\sum_{i=1}^{n_h+1}(a_{i,j}^{(h)}z_i^{(h)})\right)\right]$$
$$=\sum_{j=1}^{r}\left\{(\vec{z}^{(h)}\cdot\vec{a}_j^{(h)}+a_{0,j}^{(h)})\cdot\exp\left[-\frac{\left\|\vec{z}^{(h)}-\vec{m}_j^{(h)}\right\|^2}{\pm_j^2}\right]\right\} \quad (14)$$

Where $\vec{a}_j^{(h)}=[a_{1,j}^{(h)},\ldots,a_{n_h,j}^{(h)}]$. By setting

$$\vec{a}_j^{(h)}=w_j^{(h)}[m_{1j}^{(h)}\cdots m_{n_hj}^{(h)}]=w_j^{(h)}\vec{m}_j^{(h)} \quad (15)$$

Eq.(14) becomes

$$y(\vec{x})=\sum_{j=1}^{r}\left\{(w_j^{(h)}\vec{z}^{(h)}\cdot\vec{m}_j^{(h)}+a_{0,j}^{(h)})\cdot\exp\left[-\frac{\left\|\vec{z}^{(h)}-\vec{m}_j^{(h)}\right\|^2}{\pm_j^2}\right]\right\}$$
$$=\sum_{j=1}^{r}\left\{(w_j^{(h)}\vec{z}^{(h)}\cdot\vec{m}_j^{(h)})\cdot\exp\left[-\frac{\left\|\vec{z}^{(h)}-\vec{m}_j^{(h)}\right\|^2}{\pm_j^2}\right]\right\}$$
$$+\sum_{j=1}^{r}a_{0,j}^{(h)}\exp\left[-\frac{\left\|\vec{z}^{(h)}-\vec{m}_j^{(h)}\right\|^2}{\pm_j^2}\right] \quad (16)$$

By adopting the kernel trick, a TS-kernel is integrated as

$$K_{TS}(\vec{z}^{(h)},\vec{m}_j^{(h)})=(\vec{z}^{(h)}\cdot\vec{m}_j^{(h)})\cdot\exp\left[-\frac{\left\|\vec{z}^{(h)}-\vec{m}_j^{(h)}\right\|^2}{\pm_j^2}\right].$$

And setting $\sum_{j=1}^{r}a_{0,j}^{(h)}\exp\left[-\frac{\left\|\vec{z}^{(h)}-\vec{m}_j^{(h)}\right\|^2}{\pm_j^2}\right]=b^{(h)}$

then (16) can be further rewritten as:

$$y(\vec{x})=\sum_{j=1}^{r}w_j^{(h)}K_{TS}(\vec{z}^{(h)},\vec{m}_j^{(h)})+b^{(h)} \quad (17)$$

Eq.(17) is the output of TSK-type fuzzy system, which is equivalent to the output of the SVR.

For each level module, the parameters $\vec{m}_j$ and $\pm_j$ in the TS-kernels and the number of rules are determined by FCM clustering, the weighting parameters $w_j^{(h)}$ and bias $b^{(h)}$ are determined by the linear SVR. Each input data $\vec{z}^{(h)}$ is transformed to the vector $\vec{V}(\vec{z}^{(h)})=[V_1(\vec{z}^{(h)}),V_2(\vec{z}^{(h)}),\ldots,V_r(\vec{z}^{(h)})]$, where $V_j(\vec{z}^{(h)})=K_{TS}(\vec{z}^{(h)},\vec{m}_j^{(h)})$ is the output of the j th TS-kernel. The vector is fed as input to a linear SVR, and the training data pairs are represented by

$$S=\left\{(\vec{V}(\vec{z}_1^{(h)}),y_1^{(h)}),(\vec{V}(\vec{z}_2^{(h)}),y_2^{(h)}),\ldots,(\vec{V}(\vec{z}_N^{(h)}),y_N^{(h)})\right\} \quad (18)$$

The optimal linear SVR function is

$$y^{(h)}(\vec{z}^{(h)})=\sum_{k=1}^{N}(\sim_k^{(h)}-\hat{\sim}_k^{(h)})\left\langle V(\vec{z}^{(h)}),V(\vec{z}_k^{(h)})\right\rangle+b^{(h)} \quad (19)$$

Where $\sim_k^{(h)}$ and $\hat{\sim}_k^{(h)}$ are solved by SVR. Eq.(19) can be represented as

$$y^{(h)}(\vec{z})=\sum_{k=1}^{N}(\sim_k^{(h)}-\hat{\sim}_k^{(h)})\sum_{j=1}^{r}V_j(\vec{z}^{(h)})V_j(\vec{z}_k^{(h)})+b^{(h)}$$
$$=\sum_{j=1}^{r}\left(\sum_{k=1}^{N}(\sim_k^{(h)}-\hat{\sim}_k^{(h)})V_j(\vec{z}_k^{(h)})\right)V_j(\vec{z}^{(h)})+b^{(h)} \quad (20)$$
$$=\sum_{j=1}^{r}w_j^{(h)}V_j(\vec{z}^{(h)})+b^{(h)}=\sum_{j=1}^{r}w_j^{(h)}K_{TS}(\vec{z}^{(h)},\vec{m}_j^{(h)})+b^{(h)}$$

Where $w_j^{(h)}=\sum_{k=1}^{N}(\sim_k^{(h)}-\hat{\sim}_k^{(h)})V_j(\vec{z}_k^{(h)})$ (21)

According to Eq.(21), the weighting parameters $w_j^{(h)}$ are obtained.

3.3.2 Structure of single-level modules

The structure of each single-level FSVRN module is based on the structure like the four-layered Fuzzy neural network in Fig.3, which consists of the membership function, fuzzy rules, weighted consequent and output layers, and their functions are briefly described below in the our context. In the following descriptions, $O_{i,j}^{(h)}$ represents the output of the i th node in j th layer of level h, $u_l^{(h)}$ is the l th input of a certain node in current layer of level h.

Layer 1: Each node in this layer corresponds to a membership function of ordinary system inputs and input from previous level [see (22) and (23)].

$$O_{i,1}^{(h)}=M_{i,1}^{(h)}(x_k^{(h)}),$$
$$\forall i=1,2,\cdots,N_1^{(h)}-1,h=2,\cdots,H \quad (22)$$

and

$$O_{N_1^{(h)},1}^{(h)}=M_{N_1^{(h)},1}^{(h)}(y^{(h-1)}) \quad \forall h=2,\cdots,H \quad (23)$$

Here, $M_{i,1}^{(h)}(x_k^{(h)})$ and $M_{N_1^{(h)},1}^{(h)}(y^{(h-1)})$ are the corresponding membership value of the input (or intermediate) variable connected to it by (12). $N_1^{(h)}$ is equal to the total number of inputs to level h, including the membership value of system inputs and intermediate input. When $h=1$, there will not exist any intermediate inputs and only (22) is applied.

Layer 2: The output of each node in this layer represents the firing strength of a rule

$$O_{i,2}^{(h)}=\prod_{l=1}^{r_i}\left(u_l^{(h)}\right) \quad \forall i=1,2,\cdots,N_2^{(h)}. \quad (24)$$

The determination of (24) is similar with (13). r_i denotes the number of preconditions in rule node i and $N_2^{(h)}$ indicates the total number of rules in level h.

Layer 3: The output of each rule is computed in this layer.

In level $h=1$ where no intermediate variable appears, the function has the form

$$O_{i,3}^{(1)} = u_l^{(1)}\left(\sum_{j=0}^{n_1} a_{j,i}^{(1)} x_j^{(1)}\right) \quad \forall i = 1,2,\cdots,N_3^{(1)} \tag{25}$$

and when $h > 1$

$$O_{i,3}^{(h)} = u_l^{(h)}\left(\sum_{j=0}^{n_h+1} a_{j,i}^{(h)} z_j^{(h)}\right) \quad \forall i = 1,2,\cdots,N_3^{(h)} \tag{26}$$

where $N_3^{(1)}$ and $N_3^{(h)}$ correspond to the total number of nodes in the third layer for Level 1and Level 3,respectively. $\{a_{j,i}^{(h)}\}$is the consequent parameter set in level h , $z_0^{(h)} = 1;$.

Layer 4: The final output is computed by summing the outputs of all rules

$$O_{1,4}^{(h)} = \sum_{j=1}^{N_R^{(h)}} u_l^{(h)} \tag{27}$$

$N_R^{(h)}$ is the total number of fuzzy rules in level h , which equals to $N_2^{(h)}$. Eqs.(14), (16), (17), (18) and (20) determine the output in (27).

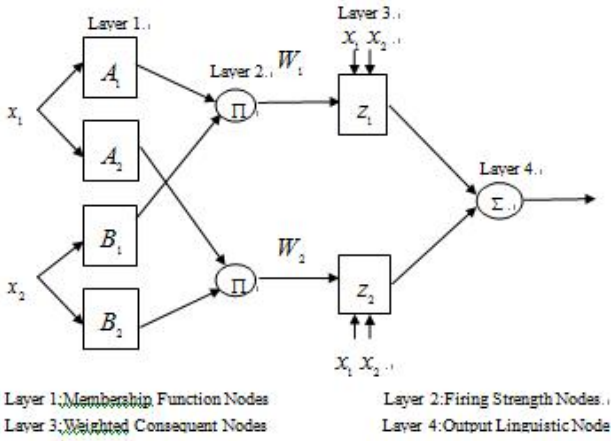


Figure 3: The structure of the four-layered network.

3.3.3 Learning algorithm

As we have found in 3.3.1, the consequent part of the TSK fuzzy inference rule is a linear combination of all consequent parameters, which can be reconstructed in the form of linear SVR. So, the linear SVR algorithm is first applied to evaluate the optimal values of all consequent parameters of TSK-type MLFSVRN model. Given a set of training data pairs and setting the desired output of each single-level module as same as the final system output. The linear SVR learning algorithm of a TSK-type MLFSVRN model with incremental architecture is listed as follows.

Step 1—Initialization:

1) Divide the input variables into H subsets $\left\{x_1^{(h)}, x_2^{(h)}, \cdots, x_{n_h}^{(h)} \middle| h = 1,\cdots,H\right\}$according to the variable selection method proposed in Section 3, each of them attached to a single-level reasoning network module.

2) Set the level index $h = 1$ and initialize appropriate membership function parameters based on the FCM clustering method. The membership function parameters of all intermediate variables are fixed according to the final outputs.

Step 2—Apply Linear SVR to Level h :

In order to evaluate the optimal value of all unknown consequent parameters $a_{j,i}^{(h)}$, (25),(26) and(27) can be rewritten in the form of linear SVR according to Eqs.(14), (16), (17), (18) and (20) . The consequent parameters then can be evaluated using Linear SVR method with (15) and (21). Here, the output value y will be used as the desired value of $y^{(h)}$.

Step 3—Forward Computation:

The output of Level h , $y^{(h)}$, can be computed using the evaluated $a_{j,i}^{(h)}$'s.

Step 4—Termination:

Set $h = h + 1$. If $h \le H$, go to Step 2; otherwise, the training process stops.

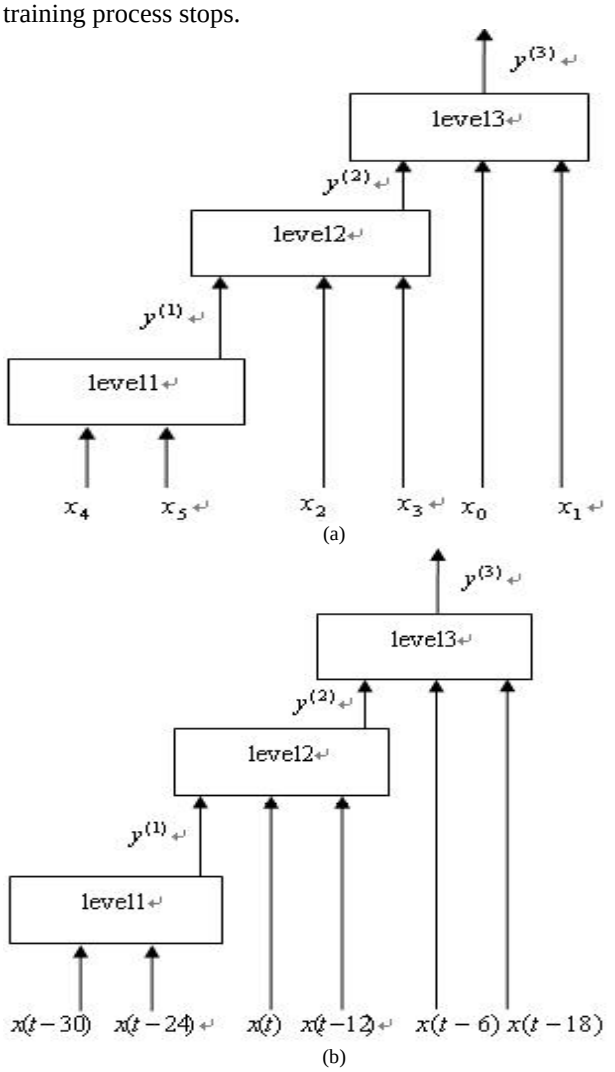


Figure 4: MLFSVRN model with incremental architecture (a) six-dimensional example (b) Mackey-Glass time series.

4 Simulation results

In this section, the proposed method has been evaluated for nonlinear system identification and Mackey-Glass chaotic time-series prediction. Section 4.1 discusses a six-dimensional example, which is used to validate the variable selection analysis method described in Sections 3. Section 4.2 discusses the Mackey–Glass chaotic time series prediction, aiming at demonstrating the satisfactory learning behavior and good generalization ability of the MLFSVRN models.

4.1 Six-dimensional example

The six-dimensional nonlinear system was given by the following equation:

$$y = (-4 + x_0^{0.5} + x_1^{-1})^2 + 5\sin(x_2 + x_3) + \exp(1 + x_4 + x_5) \quad (28)$$

A data set of 1000 pair was prepared by drawing the inputs uniformly from the six-dimensional unit hypercube. To construct the MLFSVRN with incremental architecture, using the variable selection analysis method proposed in Section 3, the input variables are grouped into three subsets $\{x_4, x_5\}$ 、 $\{x_2, x_3\}$ 、 $\{x_0, x_1\}$ .The influence degree of each input variable is evaluated and listed as follows:

x_i	x_0	x_1	x_2	x_3	x_4	x_5
ID(x_i)	0.109	0.107	0.176	0.178	0.273	0.270

It can be seen that x_4 is the most influential input among the six and this corresponds very well to what we can deduce from (28). Furthermore, the influence degree of x_0 and x_1 , x_2 and x_3 , x_4 and x_5 is about the same and this again matches with our expectation from (28). Thus, the effectiveness of the variable selection method is demonstrated by this example. Here, the threshold is chosen as T_{inc} =0.6, so the incremental architecture can have x_4 and x_5 assigned to the first level as inputs. For the other four input variables, x_2 and x_3 are put to the second level, x_0 and x_1 are put to the third level, as shown in Fig.4(a).

	Rules		Error-Train	Error-Test
TSK MLFSVRN With Incremental Architecture	Level 1	4	0.0168	0.0135
	Level 2	8		
	Level 3	8		
	Total	20		
FSVRN	48		0.0003	0.0126
Jang's ANFIS	64		0.0005	0.0157

Table 1: Comparison of the TSK-type MLFSVRN models from their single-level counterpart FSVRN and Jang's ANFIS in Function prediction.

The performance of the proposed TSK-type MLFSVRN models with incremental architecture has been evaluated with the Root Mean Square Error (RMSE). After the liner SVR learning phase with 400 training data pairs, the models were validated by the testing data set consisting of 600 points. For the purpose of comparison, the performance of the TSK-type MLFSVRN models, their single level counterpart, i.e., FSVRN model and Jang's ANFIS model [21] with 400 training data are listed in Table 1. Obviously, the TSK-type MLFSVRN models has 3 levels, in which the number of terms for each input is 2, 4, 4, respectively. The total number of rules is 20, which use much less fuzzy rules and adjustable parameters than single-level FSVRN. Furthermore, although the single-level counterparts-FSVRN has the smallest training RMSE and testing RMSE, the number of fuzzy rules is more than our proposed model. The testing RMSE of Jang's ANFIS model is biggest among these three models because it cannot have correct response to unforeseen inputs when the training samples are limited. So, the TSK-type MLFSVRN model with incremental architecture shows relatively better generalization ability.

4.2 Prediction of a chaotic time-series

The Mackey-Glass chaotic differential delay equation is recognized as a benchmark problem for time-series prediction which frequently used in the study of chaotic dynamics and defined as follows [14]:

$$\frac{dx(t)}{dt} = \frac{0.2x(t-\tau)}{1 + x^{10}(t-\tau)} - 0.1x(t) \quad (29)$$

When $\tau > 17$, the equation shows chaotic behavior. In our simulations, we set $\tau = 30$. In this paper we used $x(t-30)$, $x(t-24)$, $x(t-18)$, $x(t-12)$, $x(t-6)$ and $x(t)$ as input variables to predict the value of $x(t+6)$.

To construct the MLFSVRN with incremental architecture, using the variable selection method proposed in section 3, the input variables are grouped into three subsets $\{x(t-24), x(t-30)\}$ 、 $\{x(t-18), x(t-6)\}$ 、 $\{x(t-12), x(t)\}$ .The influence degree of each input variable is computed and listed as follows:

x_i	$x(t-30)$	$x(t-24)$	$x(t-18)$	$x(t-12)$	$x(t-6)$	$x(t)$
ID(x_i)	0.253	0.327	0.087	0.165	0.074	0.173

It is found that $x(t-30)$ and $x(t-24)$ are two most influential input variables. Among three clusters, the combination of the influence degree of $x(t-30)$ and $x(t-24)$ is biggest and less than the threshold T_{inc} =0.6 which consistently follow the algorithm in section 3. So the incremental architecture can have $x(t-30)$ and $x(t-24)$ assigned to the first level as inputs. For the other four input variables, $x(t-18)$ and $x(t-6)$ are put to the second level, $x(t-12)$ and $x(t)$ are put to the third level, as shown in Fig.4(b).

	Rules		Error-Train	Error-Test
TSK MLFSVRN With Incremental Architecture	Level 1	9	0.0253	0.0262
	Level 2	9		
	Level 3	9		
	Total	27		
FSVRN	35		0.0258	0.0352
Jang's ANFIS	39		0.0275	0.0408

Table 2: Comparison of the TSK-type MLFSVRN models from their single-level counterpart FSVRN and Jang's ANFIS in Mackey-Glass chaotic prediction.

In order to evaluate the performance of TSK-type MLFSVRN models with the Root Mean Square Error (RMSE), the 200 points of the series from $t = 501 - 700$ and a comparatively larger one consisting of 700 points of the series from $t = 130 - 829$ are used as training data, and 500 points from $t = 830 - 1329$ are used as testing data. According to the variable selection method proposed in section 3, we obtained 3 levels and the number of fuzzy rules in each level of the incremental architecture is 9. For the purpose of comparison, the performance of the proposed TSK-type MLFSVRN models, their single level counterpart, i.e., FSVRN model and Jang's ANFIS model with 700 training points are listed in Table 2. From that, it can be seen that the proposed TSK-type MLFSVRN models uses much less fuzzy rules than other models. Furthermore, it is found that the MLFSVRN models perform best among the three models in terms of training and testing RMSE. Fig.5 shows that the TSK-type MLFSVRN prediction outputs are close the real outputs and achieves good generalization ability.

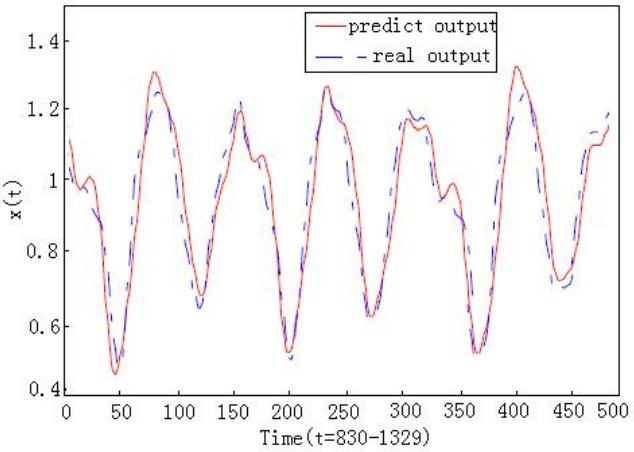


Figure 5: Test Result of the TSK-type MLFSVRN model.

The above simulations show that our proposed MLFSVRN models can get rid of the dimensionality problem fundamentally. TSK-type MLFSVRN models consume much less fuzzy rules compared with their single-level counterparts-FSVRN and Jang's ANFIS. TSK-type MSFNN models save both fuzzy rules and adjustable parameters significantly compared with Jang's ANFIS.

5 Conclusion

In this paper, a hierarchical TSK-type fuzzy system was proposed and its applications in system identification and time-series prediction were studied. In the proposed method, the major characteristic of such model is that the consequence of a rule will be used as a fact to another rule from which the number of fuzzy rules resulted will no longer be an exponential function of the number of input variables. The proposed MLFSVRN model is constructed with incremental architecture. First, some influential input variables are arranged to different reasoning levels by analyzing the influence degree of each input variable based on FCM clustering and fuzzy association rules. Then, each level reasoning module can be realized by FSVRN model. Its consequent parameters are learned by a linear SVR with a new TS-kernel. The major advantage of using MLFSVRN model other than a single-level fuzzy system is that the number of fuzzy rules and parameters involved in modelling process can be reduced significantly and the generalization ability can be improved when compared with those required by the single-level FSVRN systems and Jang's ANFIS systems. The effectiveness of the MLFSVRN model has been demonstrated through two problems. It can generally be concluded that the proposed method has higher performance in identification and time-series prediction in comparison with the other methods.

Funding

This paper is supported by the Fundamental Research Funds for the Central Universities (No.FRF-SD-12-009B) and the State Scholarship Fund.

References

[1] G. V. S. Raju, J. Zhou, and R. A. Kisner (1991), "Hierarchical fuzzy control", Int. J. Contr., vol. 54 no. 5, pp. 1201–1216.

[2] Cheong F (2007), "A hierarchical fuzzy system with high input dimensions for forecasting foreign exchange rates". IEEE Congress on Evolutionary Computation, CEC (Singapore), pp. 1642–1647.

[3] Aja-Ferna´ndez S, Alberola-Lo´pez C (2008), "Matriz modeling of hierarchical fuzzy systems", IEEE Trans Fuzzy Syst, vol. 16, no. 3, pp. 585–599.

[4] Zeng X, Goulermas J, Liatsis P, Wang D, Keane J (2008), "Hierarchical fuzzy systems for function approximation on discrete input spaces with application", IEEE Trans Fuzzy Syst, vol. 16 no. 5, pp. 1197–1215.

[5] Benftez A, Casillas J (2009), "Genetic learning of serial hierarchical fuzzy systems for large-scale problems". Proceedings of Joint 2009 International Fuzzy Systems Association World Congress and 2009 European Society of Fuzzy Logic and Technology Conference (IFSA-EUSFLAT, Lisbon), pp. 1751–1756.

[6] Zajaczkowski J, Verma B (2008), "Selection and impact of different topologies in multilayered

hierarchical fuzzy systems", Appl Intell, vol. 36, no. 3, pp. 564–584.

[7] Salgado P (2008), "Rule generation for hierarchical collaborative fuzzy system", Appl Math Modell Sci Direct, vol. 32, no. 7), pp. 1159–1178.

[8] Joo M, Sudkamp T (2009), "A method of converting a fuzzy system to a two-layered hierarchical fuzzy system and its run-time efficiency", IEEE Trans Fuzzy Sys, vol. 17, no. 1, pp. 93–103.

[9] Jelleli T, Alimi A (2010), "Automatic design of a least complicated hierarchical fuzzy system". IEEE International Conference on Fuzzy Systems (FUZZ), pp. 1–7.

[10] S. Mitra and Y. Hayashi (2000), "Neuro-fuzzy rule generation: Survey in soft computing framework", IEEE Trans. Neural Net, vol. 11, no. 3, pp. 748–768.

[11] M. F. Azeem, M. Hanmandlu, N. Ahmad (2003), "Structure Identification of Generalized Adaptive Neuro-Fuzzy Inference Systems", IEEE Trans. On Fuzzy Systems, vol. 11, no. 5, pp. 666-681.

[12] L. X. Wang (1999), "Analysis and design of hierarchical fuzzy systems", IEEE Transactions on Fuzzy systems, vol. 7, no. 5, pp. 617-624.

[13] F. L Chung and J. C. Duan (2000), "On multistage fuzzy neural network modelling", IEEE Transactions on fuzzy systems, vol. 8, no. 2, pp. 125-142.

[14] M. C. Mackey and L. Glass (1977), "Oscillation and chaos in physiological control systems", Sci., vol. 197, pp. 287–289.

[15] J. Bezdek (1981), Pattern Recognition With Fuzzy Objective Function Algorithms.New York: Plenum.

[16] W. Pedrycz (2002), "Collaborative Fuzzy Clustering", Pattern Recognition Letters, vol. 23, no. 14, pp. 1675-1686.

[17] Agrawal R, Imielinski T, Swami A (1993), "Mining association rules between sets of items in large databases". In Proceedings of the ACM SIGMOD conference on management of data, pp. 207–216.

[18] Kantardzic M,John Wiley and Sons (2003), Data mining -Concepts, models, methods, and algorithms.

[19] Y. C. Lee, T. P. Hong and W. Y. Lin (2004), "Mining fuzzy association rules with multiple minimum supports using maximum constraints". The Eighth International Conference on Knowledge-Based Intelligent Information and Engineering Systems, Lecture Notes in Computer Science, pp. 1283-1290.

[20] B. Schölkopf, A.J. Smola (2002), Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. Cambridge MA. : MIT Press.

[21] Jang, J.-S.R (1993). "ANFIS: adaptive-network-based fuzzy inference system". IEEE Transactions on systems, Man and Cybernetics, vol. 23, no. 3, pp. 665-685.

[22] Tachibana K, Furuhashi T (2002). "A structure identification method of submodels for hierarchical fuzzy modeling using the multiple objective genetic algorithm", Int J Intell Syst, vol. 17, no. 5, pp. 495–513.

[23] C. T. Lin, S. F. Liang, C. M. Yeh, and K. W. Fan (2005), "Fuzzy neural network design using support vector regression for function approximation with outliers", IEEE International Conference on Systems, Man and Cybernetics, vol. 3, pp. 2763-2768.

[24] J. M. Leski (2005), "TSK-fuzzy modeling based on -insensitive learning", IEEE Trans, Fuzzy Systems, vol. 13, no. 2, pp. 181-193.

Machine Learning Approach for Emotion Recognition in Speech

Martin Gjoreski and Hristijan Gjoreski
Department of Intelligent Systems, Jožef Stefan Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia
E-mail: {martin.gjoreski, hristijan.gjoreski}@ijs.si

Andrea Kulakov
Faculty of Computer Science and Engineering, Rugjer Boshkovikj 16, 1000 Skopje, Macedonia
E-mail: andrea.kulakov@finki.ukim.mk

Keywords: machine learning, emotions, speech, recognition, Auto-WEKA

Received: December 11, 2014

This paper presents a machine learning approach to automatic recognition of human emotions from speech. The approach consists of three steps. First, numerical features are extracted from the sound database by using audio feature extractor. Then, feature selection method is used to select the most relevant features. Finally, a machine learning model is trained to recognize seven universal emotions: anger, fear, sadness, happiness, boredom, disgust and neutral. A thorough ML experimental analysis is performed for each step. The results showed that 300 (out of 1582) features, as ranked by the gain ratio, are sufficient for achieving 86% accuracy when evaluated with 10 fold cross-validation. SVM achieved the highest accuracy when compared to KNN and Naive Bayes. We additionally compared the accuracy of the standard SVM (with default parameters) and the one enhanced by Auto-WEKA (optimized algorithm parameters) using the leave-one-speaker-out technique. The results showed that the SVM enhanced with Auto-WEKA achieved significantly better accuracy than the standard SVM, i.e., 73% and 77% respectively. Finally, the results achieved with the 10 fold cross-validation are comparable and similar to the ones achieved by a human, i.e., 86% accuracy in both cases. Even more, low energy emotions (boredom, sadness and disgust) are better recognized by our machine learning approach compared to the human.

Povzetek: Predstavljeno je prepoznavanje čustev iz govora s pomočjo strojnega učeanja.

1 Introduction and related work

Human capabilities for perception, adaptation and learning about the surroundings are often three main compounds of the definition about what intelligent behaviour is. In the last few decades there are many studies suggesting that one very important compound is left out of this definition about intelligent behaviour. That compound is emotional intelligence. Emotional intelligence is the ability of one to feel, express, regulate his own, to recognize and handle the emotional state of others. In psychology the emotional state is defined as complex state that results in psychological and physiological changes that influence our behaving and thinking [1] .

With the recent advancements of the technology and the growing research areas like machine learning (ML), audio processing and speech processing, the emotional states will be inevitable part of the human-computer interaction. There are more and more studies that are working on providing the computers with abilities like recognizing, interpretation and simulation of emotional states.

Automatic emotion recognition is part of growing research areas such as industry for robots [22], automobile industry, entertainment industry, marketing industry, and similar. The automatic emotion recognition also can be also used for improving the accuracy in speech recognition. It is expected that automatic emotion recognition will change the human-computer communication [23].

The goal of the emotion recognition systems is to recognize the emotional state that is experiencing the speaker. The focus is usually on how something is said, and not what is said. Besides the approaches where only the speaker's voice is analysed, there are several different approaches for recognizing the emotional state. In some approaches the voice and the spoken words are analysed [2] . Some are focused only on the facial expressions [3] . Some are analysing the reactions in the human brain for different emotional states [4] . Also there are combined approaches where combination of the mentioned approaches is used [5] .

In general, there are two approaches in human emotions analysis. In the first approach the emotions are represented as discrete and distinct recognition classes [6] . The other approach represents the emotional states in 2D or 3D space, where parameters like emotional distance, level of activeness, level of dominance and level of pleasure are be observed [7] .

In this research we present a ML approach for automatic recognition of emotions from speech. Our

approach uses the discrete type of approach; therefore the emotional states are represented by seven classes: anger, fear, sadness, happiness, boredom, disgust and neutral. Even though ML approaches have been proposed in the literature, our approach improves upon them by performing a thorough ML analysis, including methods for: feature extraction, feature standardization, feature selection, algorithm selection, and algorithm parameters optimization. With this analysis, we try to find the optimal ML configuration of: features, algorithms and parameters, for the task of emotion recognition in speech.

The remainder of this paper is organized as follows. Section 2 is a brief overview of speech emotion analysis. In Section 3 our ML approach for emotion recognition is presented. Section 4 presents the experimental setup and the experimental results. Finally, the conclusion and a brief discussion about the results is given.

2 Speech emotion analysis

Speech emotion analysis refers to usage of methods to extract vocal cues from speech as a marker for emotional state, mood or stress. The main assumption is that there are objectively measureable cues that can be used for predicting the emotional state of the speaker. This assumption is quite reasonable since the emotional states arouse physiological reactions that affect the process of speech production. For example, the emotional state of fear usually initiates rapid heartbeat, rapid breathing, sweating and muscle tension. As a result of these physiological activities there are changes in the vibration of the vocal folds and the shape of the vocal tract. All of this affects the vocal characteristics of the speech which allows to the listener to recognize the emotional state that the speaker is experiencing [8] .

The basic speech audio features that are used for speech emotion recognition are: fundamental frequency (human perception for fundamental frequency is pith), power, intensity (human perception for intensity is loudness), duration features (ex. rate of speaking) and vocal perturbations. The main question is: Are there any objective feature profiles of the voice that can be used for speaker emotion recognition? A lot of studies are done for the sake of providing such feature profiles that can be used for representation of the emotions, but results are not always consistent. For some basic problems like distinguishing normal speech from angry speech or distinguishing normal speech from bored speech the experimental results converge [9] . For example such converging results are showing that compared to normal speech, when expressing fear or happiness human speak with higher pitch (fundamental frequency).

Figure 1 shows an example of audio wave (top graphs) and pitch (bottom graphs) of normal speech (left) and angry speech (right). The missing parts of the pitch graphs are parts of the speech signals which would not have foundation in human perception. They relate to parameters (ex. silence threshold, voicing threshold) of the pitch analysis algorithms. The graphs show that by using the pitch as a feature, one can note the different characteristics of speech under different emotional states. On the left we have normal speech and on the right we have angry speech of the same words by the same person. On the left lower graph (normal speech) we can see that the pitch is around 120Hz and it is monotone. On the other hand the right lower graph (angry speech) we can see higher pitch (there are parts where the pitch goes up to 500Hz) and there is noticeable variability (there are parts where the pitch goes from 500Hz to 100Hz and vice versa). This simple analysis is just an example of how we can compare speech signals by using their physical characteristics. This simple approach cannot be used for speech emotion recognition. The problem arises when we have to distinguish emotional states like anger from happiness or fear from happiness. By using the basic speech audio features for describing these emotional states, the feature profiles are quite similar so distinguishing them is hard.

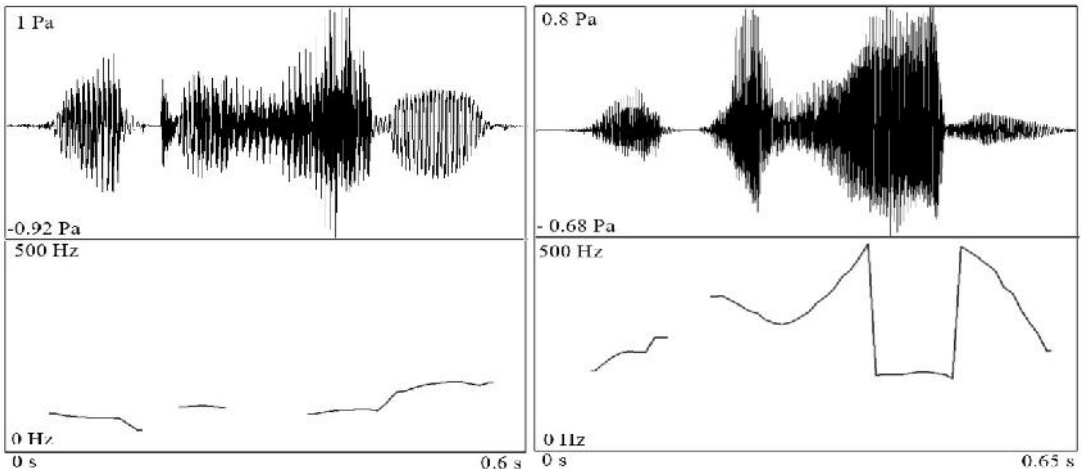


Figure 1. Audio wave (top graphs) and pitch (bottom graphs) of normal speech (left) and angry speech (right). The missing parts of the bottom graphs are parts of the speech signals which would not have foundation in human perception.

In the last few years, new method is introduced where static feature vectors are obtained by using so called acoustic Low-Level Descriptors (LLDs) and descriptive statistical functionals [10] . By using this approach a big number of large feature vectors is obtained. The downside is that not all of the feature vectors are of good value, especially not for emotion recognition. For that reason a feature selection method is often used.

3 ML Approach

Figure 2 shows the whole process of the ML speech emotion recognition used in this study. First, an emotional speech database is used, which consists of simulated and annotated utterances. Next, feature extraction is performed by using open source feature extractor. Then, feature selection method is used for decreasing the number of features and selecting only the most relevant ones. Finally, the emotion recognition is performed by a classification algorithm.

3.1 Emotional speech database

There are several emotional speech databases that are extensively used in the literature [11] : German, English, Japanese, Spanish, Chinese, Russian, Dutch etc. One main characteristic of an emotional speech database is the type of the emotions expressed in the speech: whether they are simulated or they are extracted from real life situations. The advantage of having a simulated speech is that the researcher has a complete control over the emotion that it is expressed and complete control over the quality of the audio. However, the disadvantage is that there is loss in the level of naturalness and spontaneity. On the other hand, the non-simulated emotional databases consist of a speech that is extracted from real life scenarios like call-centers, interviews, meetings, movies, short videos and similar situations where the naturalness and spontaneity is kept. The disadvantage is that in these databases there is not a complete control over the expressed emotions. Also the low quality of the audio can be problem.

For this research the Berlin emotional speech database [12] is used, which is one of the most exploited databases for speech emotion analysis. It consists of 535 audio files, where 10 actors (5 male and 5 female) are pronouncing 10 sentences (5 short and 5 long). The sentences are chosen so that all 7 emotions that we are analyzing can be expressed. The database is additionally checked for naturalness by testing it with 20 human volunteers. The volunteers were supposed to recognize and rate the naturalness of the expressed emotion by listening to random utterance. The utterances that were rated with more than 60% naturalness and from which the expressed emotion was recognized with more than 80%, were included in the final database. In Figure 3 statistics for the Berlin emotional speech database is shown. We can see information about the number of instances per class and information about the human recognition rate obtained from the tests.

Table 1. Statistics for the Berlin emotional speech database, including the human recognition rate.

Emotions	Number of instances	Human recognition rate (%)
Anger	127	96.2
Neutral	79	88.2
Fear	69	87.3
Boredom	81	86.2
Happiness	71	83.7
Sadness	62	80.7
Disgust	46	79.6

3.2 Feature Extraction

The feature extraction tool used in this research is OpenSmile (Open Speech and Music Interpretation by Large Space Extraction) [13] . It is a commonly used tool for signal processing and feature extraction when ML approach is applied on sound data. OpenSmile provides configuration files that can be used for extracting predefined features. For this research the configuration file ‘emobase2010’ is used. By using the ‘emobase2010’ configuration file in total 1582 features are extracted [14] . OpenSmile computes LLDs from basic speech features (pitch, loudness, voice quality) or representations of the speech signal (cepstrum, linear predictive coding). On these LLDs functionals are applied and static feature vectors are computed, therefore static classifiers can be used. The functionals that are applied are: extremes (position of mix/min value), statistical moments (first to forth), percentiles (ex. the first quartile), duration (ex. percentage of time the signal is above threshold) and regression (ex. the offset of a linear approximation of the contour)

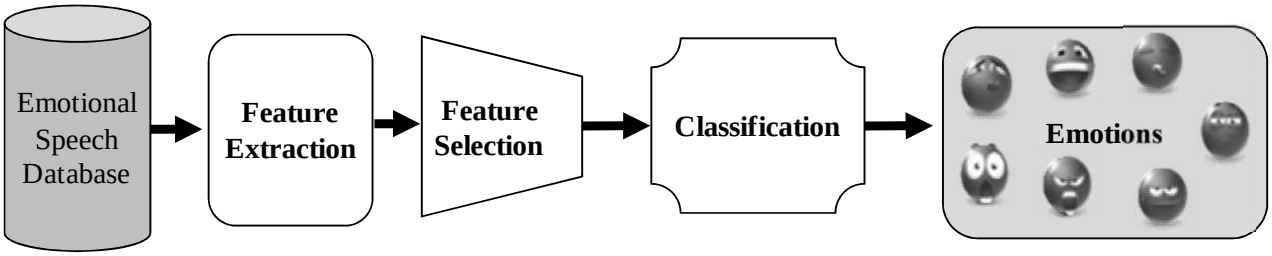


Figure 2. ML approach for emotion recognition.

After the feature extraction the feature vectors are standardized so the distribution of the values of each feature is with mean equal to 0 and standard deviation equal to 1. This way, the values for each feature are on the same scale from -1 to 1, preventing some features (with bigger values) to have more influence when creating the ML model. This is an important step in ML, especially for classification algorithms that do not have mechanism for feature standardization.

3.3 Feature Selection

Feature selection is the process of selecting a subset of relevant features for use in model construction. The central assumption when using a feature selection technique is that the data contains many redundant or irrelevant features. Redundant features are those which provide no more information than the currently selected features, and irrelevant features provide no useful information in any context [15] .

To deal with this issue, we used a method for feature selection. Features were ranked with an algorithm for feature ranking and experiments were performed with varying number of top ranked features. For ranking the features the well-known gain ratio [16] algorithm is used. Gain ratio is the ratio of information gain and the entropy of one feature. It is used to avoid overestimation of multi-valued features (the drawback of information gain).The algorithm is used as it is implemented in Orange ML toolkit [18] .

3.4 Classification

Once the features are extracted, standardized and selected, they are used to form the feature vector database. Each data sample in the data base is an instance, i.e., feature vector, used for classification. Because each instance is labeled with the appropriate emotion, supervised classification algorithms are used. In our experiments three commonly used algorithms for classification were tested, K-Nearest Neighbors (KNN) [19] , Naïve Bayes [21] and Support Vector Machine (SVM) [20] . KNN is an instance-based classifier (lazy) that does not learn a model, but it uses similarity metrics (e.g. Euclidian distance) to find the K most similar training instances and apply the majority class value (emotion) of these K instances. Naïve Bayes is a probability-based algorithm. It applies a probability theory to the feature values in order to create a model that divides the instances according to the class values. The SVM is the most complex of the three algorithms. Its goal is to find hyperplanes in the attribute's space in order to maximize the margin between instances that belong to distinct classes. It uses a kernel function in order to create non-linear classifiers.

We performed thorough experiments with each of the classification models, and once we selected the one with the highest recognition accuracy, we further enhanced its accuracy with Auto-WEKA [25] . Auto-WEKA is a ML tool that is using approach for parameter optimization of classification algorithms. It searches to the huge space of algorithm parameters and by using an intelligent optimization functions finds the near optimal parameter setting, which should increase the accuracy of the chosen algorithm. The problem of parameter optimization is viewed as a single hierarchical parameter optimization in which even the classification algorithm is considered as a parameter. The root-level parameter is the learning algorithm and the rest of the searching space is depending on the chosen parameter (algorithm) in the previous level. The main idea of Auto-WEKA is that search in the combined space of algorithms and parameters results with better-performing models than standard algorithm selection and parameter optimization methods. For searching the huge space of parameters Auto-WEKA is using Bayesian optimization methods TPE [23] and SMAC [24] .

For the model selection problem Auto-WEKA splits the train data in k folds so that each of the pre-selected learning algorithms is tested with k-fold cross validation. The ultimate goal of the model selection approach is to find an algorithm with optimal generalization performance. The selection criteria is minimizing the misclassification rate. For the parameter optimization problem Auto-WEKA is using hierarchical parameter search and again the goal of the optimization problem is minimization of the misclassification rate.

4 Experiments

4.1 Experimental setup

The experiments were conducted in the following order. First, the OpenSmile feature extraction tool extracted 1582 feature. Next, all of the extracted features were ranked using the gain ratio algorithm [16] . Then, we tested the accuracy of the three algorithms by using different number of features as ranked by the ranking algorithm. In particular we used 50, 100, 200, 400, 500, 600, 750, 1000, and 1582. As an evaluation metrics, we used the 10 fold cross-validation, which is a gold-standard technique for evaluating datasets when the instances are not structured or time dependent (e.g., time series). It usually gives a good estimate of the accuracy of an algorithm.

In the next step, the algorithm with the highest accuracy is further evaluated with Leave-One-Speaker-Out (LOSO) technique. Because in our case the data are collected by 10 individuals (speakers) the model was trained on the data recorded for nine people and tested on the remaining person. This procedure was repeated for each person (10 times). The LOSO evaluation approach is more reliable than using the same person’s data for training and testing if the model is intended to be used by unknown people (not included in the training dataset).

Finally, we applied Auto-WEKA toolkit in order to optimize the parameters of the chosen algorithm, i.e., parameters, and therefore to improve the accuracy. For each Auto-WEKA experiment the SMAC optimization method was chosen. The optimization timeout for Auto-WEKA was set to 24h. The training memory per experiment was set to 1000MB and the training run

timeout was set to 150 min. For these experiments the Auto-WEKA's feature selection module was turned off.

For each comparison, tests to confirm the statistical significance of the results were performed using paired Student's T-test with a significance level of 5%.

Three, commonly used in ML, evaluation metrics were analyzed: the recall, precision and accuracy. The following formulas define each of the metrics, where Q can be any emotion that we are trying to recognize (happiness, neutral, etc.):

$$recall = \frac{\text{No. of correctly recognized emotions labeled as Q}}{\text{No. of all the emotions labeled as Q}} \quad (1)$$

$$precision = \frac{\text{No. of correctly recognized emotions labeled as Q}}{\text{No. of all the emotions recognized as Q}} \quad (2)$$

$$accuracy = \frac{\text{No. of correctly recognized emotions of all types}}{\text{No. of all the emotions}} \quad (3)$$

4.2 Experimental Results

Once the features were extracted and ranked by the gain ratio algorithm, we compared the accuracy of the three ML algorithms (KNN, SVM and Naïve Bayes) for different number of top-ranked features (50, 100, 200, 400, 500, 600, 750, 1000, and 1582). The results presented in Figure 3 show that, as the number of features increases up to 400, also the accuracy increases for each of the three algorithms. After that, the accuracy drops for the SVM, and small (statistically insignificant) improvements are noticed for the other two algorithms. The decrease in performance as the number of features increases is due to overfitting. This is especially notable for the SVM, which was in a way expected because its model is more complex compared to the KNN and Naive Bayes and this complexity usually increases as the number of features increases. If too many not relevant features are used, it will overfit on the training data and the accuracy on the test data will drop (which is the case in our experiments).

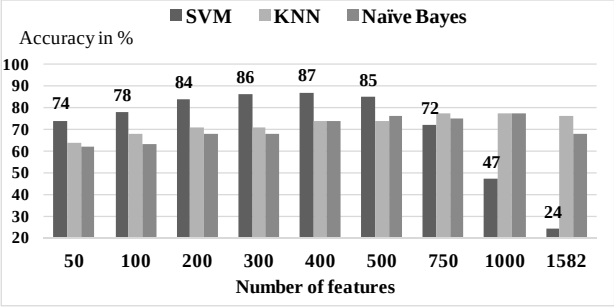


Figure 3. SVM, KNN and Naïve Bayes accuracy for 10 fold-cross validation with varying number of features.

We additionally analysed the results obtained by the SVM, which achieved the highest accuracy, i.e., 87% when the top ranked 400 features are used. However, there was no statistical difference between the accuracy achieved for 300 and 400 features. Therefore, for further analysis we used the top ranked 300 features, which was a good tradeoff between accuracy and number of features.

Figure 4 shows the confusion matrix of the SVM for the top ranked 300 features. Additionally we present the recall and the precision for each class (emotion), and the overall accuracy. The highest precision and recall are achieved for the class "sadness" and the lowest are achieved for the class "happiness". Also we can see that the classes "anger" and "happiness" are often mixed by the classifier. The class "fear" is mixed with all other 6 classes.

10 fold cross-valid. Conf. Matrix SVM		PREDICTED CLASS							Recall (%)
		A	B	D	F	H	N	S	
REAL CLASS	Anger (A)	**116**	0	1	0	10	0	0	91
	Boredom (B)	0	**72**	3	1	0	3	2	89
	Disgust (D)	2	2	**40**	1	0	1	0	87
	Fear (F)	3	1	1	**53**	4	6	1	77
	Happiness (H)	17	0	0	2	**51**	1	0	72
	Neutral (N)	0	6	0	2	1	**69**	1	87
	Sadness (S)	0	0	0	0	0	1	**61**	98
Precision (%)		84	89	89	90	77	85	94	**Acc = 86%**

Figure 4. Confusion matrix for SVM obtained with 10 fold cross-validation with the top ranked 300 features.

In the next step, we evaluated the SVM algorithm with LOSO technique. We compared the accuracy of the standard SVM (with default parameters) and the one enhanced by the Auto-WEKA. The results are shown for each test subject (speaker) individually in Figure 5. The results show that the SVM enhanced with Auto-WEKA achieved significantly better accuracy than the standard SVM, except for the first two speakers (S1 and S2).

Also we can see that the accuracy depends on test subject. For example by using Auto-WEKA the lowest average accuracy (64%) is obtained when the speaker S2 is used as a test speaker and the highest average accuracy (91%) is obtained when the user S8 is used as test speaker.

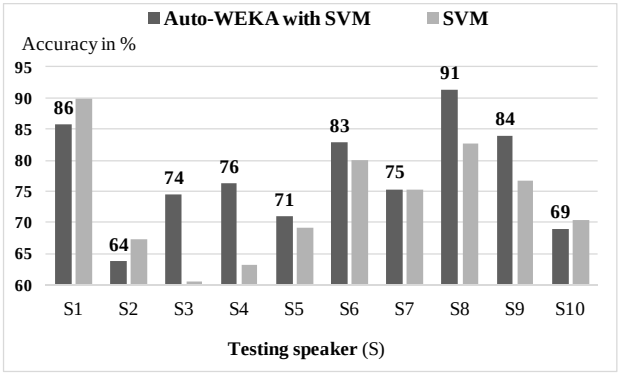


Figure 5. Auto-WEKA and SVM classification accuracy for LOSO with top ranked 300 features

Figure 6 shows more detailed analysis, i.e., confusion matrix of the results achieved by the SVM enhanced with Auto-WEKA. The highest precision and recall are achieved for the class "sadness" and the lowest are achieved for the class "happiness". Also we can see that the class "anger" is mixed with the class "happiness" and vice versa. Also the class "boredom" is mixed with the class "disgust". For the classes "fear" and "neutral"

there is no single class that can be pointed as mixing class. These classes are mixed with several others.

LOSO Conf. Matrix Auto-weka		PREDICTED CLASS							Recall (%)
		A	B	D	F	H	N	S	
REAL CLASS	Anger (A)	**105**	0	1	1	20	0	0	83
	Boredom (B)	0	**63**	11	0	0	4	3	78
	Disgust (D)	2	1	**38**	1	1	3	0	83
	Fear (F)	7	1	3	**46**	3	8	1	67
	Happiness (H)	22	0	0	3	**44**	2	0	62
	Neutral (N)	0	7	4	4	4	**59**	1	75
	Sadness (S)	0	3	0	0	0	1	**58**	94
Precision (%)		77	84	67	84	61	77	92	Acc = 77%

Figure 6. Confusion matrix for Auto-WEKA obtained with LOSO cross-validation with top ranked 300 features.

Finally, we compared the recognition accuracy achieved by a human (manual recognition) and the two ML techniques: SVM trained and tested with 10 fold cross-validation and SVM trained and tested with LOSO (the results are shown in Figure 7). The human recognition rate is obtained from the tests for checking the naturalness of the database. That is, 20 volunteers were asked to recognize the emotions from the audio files. The results show that the SVM trained and tested with 10 fold cross-validation achieved similar results as the human; on average, both achieve 86% accuracy. However, low energy emotions (boredom, sadness and disgust) are better recognized by ML compared to human. This means that for these emotions, the human ear requires additional information, which can be easily extracted using a software tools. The comparison to the LOSO shows that a model trained on subjects different from the ones used for testing should achieve significantly lower accuracy, i.e., 77%.

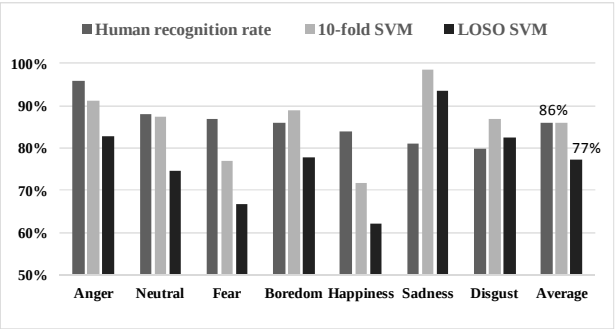


Figure 7. Comparison between the recognition accuracy achieved by a human, and the two ML techniques: 10 fold cross-validation and LOSO.

5 Conclusion

Even though ML approaches have been proposed in the emotion recognition literature, our approach improves upon them by performing a thorough ML analysis, including methods for: feature extraction and standardization, feature selection analysis, algorithm selection analysis, and algorithm parameters optimization. With this whole analysis, we tried to find the optimal ML configuration of: features, algorithms and parameters, for the task of emotion recognition in speech.

The results showed that the top 300 features, as ranked by the gain ratio, are sufficient for achieving 86% accuracy in emotion recognition. Adding more would just cause overfitting. SVM achieved the highest accuracy and significantly outperformed the KNN and Naive Bayes.

When trained and tested with 10 fold cross-validation, SVM achieved 86% over all the emotions. The per-emotion analysis shows that the highest precision and recall were achieved for the “sadness” and the lowest were achieved for the “happiness”. Also the “anger” and “happiness” were often mixed by the classifier, and the “fear” was mixed with all other 6 emotions.

In the next step, we evaluated the SVM algorithm with LOSO technique. We compared the accuracy of the standard SVM (with default parameters) and the one enhanced by the Auto-WEKA. The results showed that the SVM enhanced with Auto-WEKA achieved significantly better accuracy than the standard SVM. The overall accuracy achieved was 73% and 77% for the standard SVM and the one enhanced with Auto-WEKA. The highest accuracy (94%) was achieved for the “sadness” emotion and the lowest accuracy (62%) for the “happiness”. The confusion matrix in Figure 6 shows that the “anger” is mixed with the “happiness” and vice versa. Also the “boredom” is mixed with “disgust”.

SVM trained and tested with 10 fold cross-validation achieves better accuracy compared to LOSO, 86% and 77% accuracy respectively. The reason for this is that with the 10 fold cross-validation the training and the testing data usually contain data samples of the same speaker. On the other hand, the LOSO technique gives better estimate if the system for speech emotion recognition is supposed to work in an environment where it does not have any information about the speaker. A hybrid approach that includes a calibration phase at the beginning of the usage of the system (for example asking the user to record several data samples) is considered for future work.

The recognition accuracy achieved by the SVM trained and tested with 10 fold cross-validation is similar to the one achieved by human (manual recognition); in both cases the accuracy is 86%. Even more, low energy emotions (boredom, sadness and disgust) are better recognized by ML compared to the human. This means that for these emotions, the human ear requires additional information, which can be easily extracted using a software tools.

Auto-WEKA is state of the art approach for parameter optimization in ML. This is the first time that it is used in the field of speech analysis especially in speech emotion recognition where there is not yet gold standard approach that is widely accepted by the research community.

For future work we plan to test our approach on other languages and to provide language independent model for emotion recognition. This is possible since the emotions that we are trying to recognize are proven to be

universal and the features that we are using are language-independent. The ultimate goal would be real time language independent emotion recognition service that can be used as a part of a human affect tracking system which promotes wellbeing.

References

[1] D. G. Myers. Theories of Emotion. Psychology: Seventh Edition. New York NY: Worth Publishers. 2004.

[2] V. Perez-Rosas, R. Mihalcea. Sentiment Analysis of Online Spoken Reviews. Interspeech, 2013.

[3] A. Halder, A. Konar, R. Mandal, A. Chakraborty. General and Interval Type-2 Fuzzy Face-Space Approach to Emotion Recognition. IEEE Transactions on Systems, Man, and Cybernetics,43 (3), 2013.

[4] R. Horlings, D. Datcu, L. J. M. Rothkrantz. Emotion recognition using brain activity. Proceeding CompSysTech '08 Proceedings of the 9th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing, 2008.

[5] A. Metallinou, S. Lee, S. Narayanan. Audio-Visual Emotion Recognition Using Gaussian Mixture Models for Face and Voice. Multimedia. 2008. ISM 2008. IEEE International Symposium on Multimedia, 2008.

[6] P. Ekman. Emotions in the Human Faces. 1982.

[7] James A. Russell. A circumplex model of affect. 1980.

[8] P. N. Juslin, K. R. Scherer. Vocal expression of affect. In J. A. Harrigan, R. Rosenthal, & K. R. Scherer (Eds.). The new handbook of methods in nonverbal behavior research, pp. 65-135, 2004.

[9] K. R. Scherer. Vocal communication of emotion: A review of research paradigms. Speech Communication 40: 227–256. 2003

[10] M. E. Mena. Emotion Recognition From Speech Signals, 2012.

[11] D. Ververidis, C. Kotropoulos. A review of emotional speech databases. In: PCI 2003. 9th Panhellenic Conference on Informatics., pp. 560–574, 2003.

[12] F. Burkhardt, A. Paeschke, M. Rolfes, W. Sendlmeier, B. Weiss. A Database of German Emotional Speech. 2005. In: Proc. Interspeech. pp. 1517–1520.

[13] F. Eyben, M. Wöllmer, B. Schuller. openSMILE - The Munich Versatile and Fast Open-Source Audio Feature Extractor. 2010.

[14] F. Eyben, F. Weninger, M. Wollmer, Bjorn Schuller. openSmile Documentation. Version 2.0.0., 2013.

[15] G. Isabellem E. André. "An Introduction to Variable and Feature Selection". JMLR, 2003.

[16] H. Deng, G. Runger, E. Tuv. Bias of importance measures for multi-valued attributes and solutions. Proceedings of the 21st International Conference on Artificial Neural Networks (ICANN2011). 2011

[17] I. Kononenko, E. Simec, M. Robnik-Sikonja. Overcoming the myopia of inductive learning algorithms with RELIEFF. Applied Intelligence, Forthcoming.

[18] J. Demšar, B. Zupan. Orange: From experimental machine learning to interactive data mining. White Paper (http://www.ailab.si/orange). Faculty of Computer and Information Science. University of Ljubljana.

[19] D. Aha, D. Kibler. Instance-based learning algorithms. 1991, Machine Learning. 6:37-66.

[20] N. Cristianini, J. Shawe-Taylor. An Introduction to Support Vector Machines and other kernel-based learning methods. Cambridge University Press, 2000.

[21] R. Stuart, N. Peter. Artificial Intelligence: A Modern Approach. Second Edition, Prentice Hall.

[22] I. Witten and E. Frank, Data Mining: Practical Machine Learning Tools and Techniques, 2nd edn, Morgan Kaufmann, 2005.

[23] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl. Algorithms for Hyper-Parameter Optimization. In Proc. Of NIPS-11, 2011.

[24] F. Hutter, H. Hoos, and K. Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In Proc. of LION-5, pages 507–523, 2011.

[25] Auto-WEKA: Combined Selection and Hyperparameter Optimization of Classifiaction Algorithms. Chris Thornton, Frank Hutter, Holger Hoos, and Kevin Leyton-Brown. In Proc. of KDD 2013, 2013.

The Inherent Context Awareness of Natural User Interfaces: a Case Study on Multitouch Displays

Bojan Blažica
XLAB Research, Pot za Brdom 100, 1000 Ljubljana, Slovenia
Jožef Stefan International Postgraduate School, Jamova 39, 1000 Ljubljana, Slovenija
E-mail: bojan.blazica@xlab.si
www.xlab.si

Thesis Summary

Keywords: natural user interfaces, context-awareness, multitouch displays, biometric identification

Received: November 11, 2014

In computer science, context-awareness refers to the capability of a computing device to sense, understand and react to contextual information, i.e. information that is not at the centre of an activity but is still relevant for that activity. A computing device does not necessarily interact with humans at a given moment, but when it does, its Context-Awareness has many implications for human-computer interaction. The thesis that this paper surmises looks at Natural User Interfaces from a Context-Awareness perspective.

Povzetek: V računalništvu se pojem kontekstna ozaveščenost nanaša na sposobnost računalniškega sistema, da zazna, razume in se odzove na informacije, ki izvirajo iz konteksta, v katerem se nahaja in deluje. Imenujemo jih kontekstne informacije in jih definiramo kot tiste informacije, ki sicer niso v centru neke aktivnosti, a so zanjo še vedno pomembne. V primeru, da računalniški sistem interagira s človekom, ima lahko kontekstna ozaveščenost sistema velik vpliv na samo komunikacijo človek-računalnik. Ta članek povzema disertacijo, ki z vidika kontekstne ozaveščenosti obravnava naravne uporabniške vmesnike.

1 Introduction

This paper surmises a PhD thesis [1] that looks at natural user interfaces from a context-awareness perspective (Figure 1). On the one hand, we show that considering natural user interfaces as context-aware systems further increases the expressive power of these interfaces and, on the other hand, we show that natural user interfaces can also represent essential building blocks for context-aware systems and are therefore a viable way towards context-awareness. Research prospects addressed that arise from this perspective are: to what extent are natural user interfaces already inherently context-aware, how to increase the expressiveness of natural user interfaces through context-awareness, do natural user interfaces provide enough information to perform biometric user identification, and how to take advantage of information implicitly conveyed by the user during interaction with natural user interfaces. The specific natural user interfaces used are multitouch displays.

The thesis first reviews the fields of natural user interfaces and context-awareness. Regarding natural user interfaces, as this is an emerging research field, special care is taken to survey all currently available definitions of the term. Similarly, multitouch displays and multitouch interaction are described in more detail as they are considered in the case studies for the thesis. Other related fields such as ubiquitous/pervasive computing, ambient intelligence etc. are also briefly explained. The presented overview does not merely introduce the topic of the thesis, but also shows how interconnected these fields are and how natural user interfaces are indeed inherently context-aware.

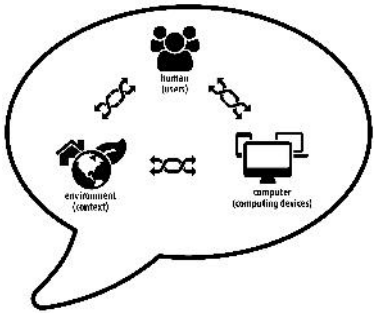


Figure 1: Context in human-computer interaction.

2 MTi: a Method for user identification on multitouch displays

We have shown how the increased amount and variety of data from natural user interfaces can be exploited to acquire contextual information by developing a biometric user identification method and a clustering algorithm for hand detection, both for multitouch displays. The method

for user identification, named MTi, is based on features obtained only from the coordinates of the 5 touchpoints of one of the user's hands (Figure 2). This makes it applicable to (almost) all multitouch displays without requiring additional hardware and regardless of the display's underlying sensing technology. The method was tested on a dataset of 34 users and reported 94.69 % identification accuracy. The method also proved to scale well and has an above-average usability [2].

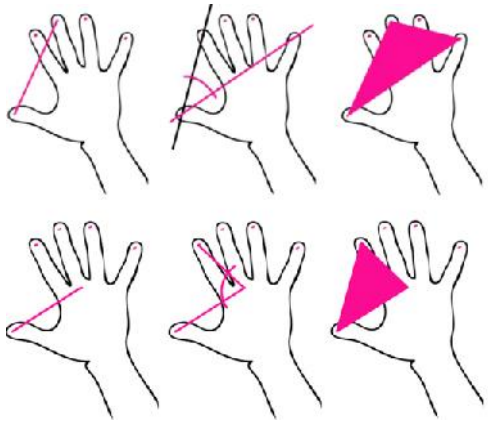


Figure 2: Illustration of types of features used for user identification on multitouch displays.

3 HDCMD: a clustering algorithm to support hand detection on multitouch displays

Next, we address the problem of hand detection, i.e. detecting how many hands are currently on the surface and associating each touch point to its corresponding hand (Figure 3) [3]. The presented solution – a clustering algorithm with simple heuristics based on the anatomy of the human hand – is software-based and thus again applicable to all multitouch surfaces regardless of their construction. Along with these two, other related methods that increase the expressiveness of multitouch displays are surveyed in [2, 3].

4 A personal perspective on photowork: implicit human computer interaction for photo collection management

Finally, the thesis explores the possibility to use implicit human-computer interaction to aid personal photo collection management. The idea is that the way we interact with natural user interfaces can implicitly disclose additional (contextual) information, which helps a context-aware system to better understand the user. More specifically, we take into account the user's personal relationship with a single photo; whether the photo is of particular importance to the user. We call this personal relationship the user's affinity for a photo.

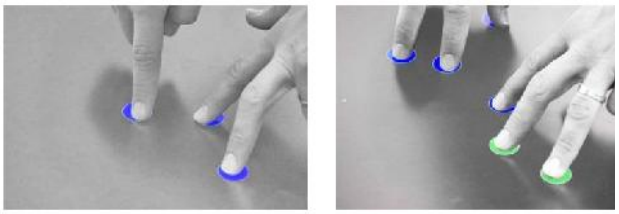


Figure 3: Illustration of the problem of hand detection on multitouch displays.

Experiments revealed that affinity is correlated with the time a user spends viewing a picture. Furthermore, by looking at viewing times, it is also possible to distinguish the task a user is currently performing [4].

5 Conclusion

The positive examples of context acquisition on multitouch displays presented confirm that natural user interfaces are inherently context-aware and show how their expressive power can be further increased by viewing them from a context-aware perspective.

Acknowledgement

The author wants to thank his supervisors Dunja Mladenić and Daniel Vladušič for their advice and support. Funded in part by the European Union, European Social Fund, Operational Program for Human Resources, Development for the Period 2007-2013 and by the Slovenian Research Agency.

References

[1] Blažica, B. The inherent contex awareness of natural user interfaces: a case study on multitouch displays : doctoral dissertation. Ljubljana, 2013 http://hci.si/?p=511

[2] Blažica, B.; Vladušic, D.; Mladenic, D. Mti: A method for user identification for multitouch displays. International Journal of Human-Computer Studies vol. 71, no. 6, p. 691–702 (2013). http://goo.gl/pcyUzl.

[3] Blažica, B.; Vladušic, D.; Mladenic, D. HDCMD: a Clustering Algorithm to Support Hand Detection on Multitouch Displays. V: First International Conference, SouthCHI 2013, Maribor, Slovenia, July 1-3, 2013. Human factors in computing and informatics : proceedings (Lecture notes in computer science), vol. 7946, p. 803-814 (2013).

[4] Blažica, B.; Vladušic, D.; Mladenic, D. A personal perspective on photowork: implicit human-computer interaction for photo collection management. Personal and Ubiquitous Computing, vol. 17, no. 8, p. 1787-1795 (2013). http://goo.gl/sGfvNh.

The Role Of Hubness in High-dimensional Data Analysis

Nenad Tomašev
Artificial Intelligence Laboratory, Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia
E-mail: nenad.tomasev@gmail.com

Thesis Summary

Keywords: machine learning, curse of dimensionality, hubness

Received: November 9, 2014

This article presents a summary of the doctoral dissertation of the author, which addresses the task of machine learning under hubness in intrinsically high-dimensional data.

Povzetek: Prispevek predstavlja povzetek doktorske disertacije avtorja, ki obravnava naloge strojnega učenja pri zvezdiščnosti visokodimenzionalnih podatkov.

1 Introduction

Machine learning in intrinsically high-dimensional data is known to be challenging and this is usually referred to as the curse of dimensionality. Designing machine learning methods that perform well in many dimensions is critical, since high-dimensional data arises often in practical applications and typical examples include textual, image and multimedia feature representations, as well as time series and biomedical data.

The hubness phenomenon [1] has recently come into focus as an important aspect of the curse of dimensionality that affects many instance-based machine learning systems. With increasing dimensionality, the distribution of instance relevance within the models tends to become long-tailed. A small number of hub points dominates the analysis and influences a disproportionate number of system predictions. Most remaining points are rarely or never retrieved in relevance queries, resulting in an information loss. High data hubness has been linked to poor system performance in many data domains.

The dissertation [2] proposes several novel hubness-aware machine learning algorithms to improve the effectiveness of machine learning in intrinsically high-dimensional data. The proposed methods are based on modeling the influence of hub points on the training data.

The article is organized as follows. Section 2 gives an overview of the proposed hubness-aware methods. Section 3 summarizes the evaluation results. The dissertation's scientific contributions are outlined in Section 4 together with plans for future work.

2 Hubness-aware machine learning

2.1 Classification

The earlier approach of using hubness-based weighting in k-nearest neighbor classification was extended and several hubness-aware classification algorithms based on class-conditional k-nearest neighbor occurrence models were proposed: h-FNN, dwh-FNN, HIKNN and NHBNN. The fuzzy hubness-aware k-nearest neighbor methods (h-FNN, dwh-FNN) utilize hubness-based fuzzy measures for voting in an extended fuzzy k-nearest neighbor (FNN) framework. HIKNN further extends dwh-FNN by taking the neighbor occurrence self-information into account and assigning higher relevance to less frequently occurring neighbor points. HIKNN also removes the need for special anti-hub handling mechanisms and reduces the number of parameters needed for dwh-FNN. Unlike the fuzzy approaches, the naive hubness-Bayesian k-nearest neighbor method (NHBNN) presents a Bayesian re-interpretation of the neighbor occurrence events and offers a novel probabilistic framework for kNN classification.

2.2 Clustering

It was demonstrated in the dissertation that the neighbor occurrence frequencies in intrinsically high-dimensional data tend to be correlated with local cluster centrality. This was used to propose several deterministic and stochastic extensions of the well-known K-means clustering framework: local K-hubs (LKH), global K-hubs (GKH), local hubness-proportional clustering (LHPC), global hubness-proportional clustering (GHPC) and global hubness-proportional K-means (GHPKM) [3]. The novelty of the methods is that they exploit hubs as cluster prototypes and use point-wise hubness for guiding the search for the optimal configuration.

2.3 Metric learning

A hubness-aware extension of the commonly used $simcos_s$ shared-neighbor secondary similarity measure was proposed, $simhub_s$. The proposed approach uses hubness-based weights when calculating the secondary similarity

scores.

3 Evaluation

The proposed classification, clustering and metric learning approaches were evaluated on many intrinsically high-dimensional datasets from various domains, as well as specially generated challenging synthetic datasets. They were compared with standard hubness non-aware baselines and statistically significant improvements were observed.

The proposed approaches were shown to improve system performance in several highly challenging tasks, including learning under class imbalance and learning with feature or label noise. A disproportionate amount of misclassification in high-dimensional class-imbalanced data was determined to be caused by minority class points. Hubness-aware classifiers were found to be well suited for classification under this newly discovered *curse of minority hubs* [4].

4 Conclusions

The dissertation addresses the problem of designing effective machine learning approaches under the assumption of hubness in intrinsically high-dimensional data. It proposes several novel hubness-aware methods for learning in many dimensions. The main contributions of the dissertation are:

- Novel hubness-aware kNN classification methods: h-FNN, dwh-FNN, HIKNN and NHBNN. The novelty lies in using class-conditional neighbor occurrences for hub modeling on the training data.
- Novel clustering approaches: LKH, GKH, LHPC, GHPC, GHPKM. These are the first hubness-based clustering approaches to be proposed.
- A novel secondary similarity measure, $simhub_s$. It is the first shared-neighbor similarity score to take hubness explicitly into account.

The proposed hubness-aware approaches achieved statistically significant improvements over the corresponding baselines in various tested experimental contexts.

As future work, we wish to further extend the proposed hubness-aware approaches. We intend to pay special attention to unsupervised and semi-supervised learning, as obtaining ground truth is often expensive in practical applications. We also intend to work on scalability in order to make the proposed approaches useful on large-scale datasets.

References

[1] M. Radovanović (2011) *Representations and Metrics in High-Dimensional Data Mining*, Izdavačka knjižarnica Zorana Stojanovića, Novi Sad, Serbia.

[2] N. Tomašev (2013) *The Role of Hubness in High-dimensional Data Analysis*, IPS Jož Stefan, Ljubljana, Slovenia.

[3] N. Tomašev et al. (2013) The Role of Hubness in Clustering High-dimensional data, *IEEE Transactions on Knowledge and Data Engineering*, IEEE Computer Society, pp. 1041–4347.

[4] N. Tomašev et al. (2013) Class Imbalance and The Curse of Minority Hubs, *Knowledge-Based Systems*, Elsevier, pp. 157–172.

CONTENTS OF *Informatica* Volume 38 (2014) pp. 1–391

Papers

MIRCHEVSKA, V. & . 2014. Classifier Generation by Combining Domain Knowledge and Machine Learning . Informatica 38:91–92.

MOSTAFA, M. &, E.M.F. EL HOUBY, A. SALAH. 2014. Semantic Searching of Biological Documents Using Gene Ontology. Informatica 38:71–80.

MUNEZERO, M. &, C.S. MONTERO, T. KAKKONEN, E. SUTINEN, M. MOZGOVOY, V. KLYUEV. 2014. Automatic Detection of Antisocial Behaviour in Texts. Informatica 38:3–10.

PAN, W. &, C. YAO, X. LI, L. SHEN. 2014. An Online Compression Algorithm for Positioning Data Acquisition. Informatica 38:339–346.

PAROL, P. &, M. PAWLOWSKI. 2014. Future Proof Access Networks for B2B Applications. Informatica 38:193–204.

PURGINA, M. &, A. KUZNETSOV, E. PYSHKIN. 2014. Leveraging User Experience through Input Style Transformation to Improve Access to Music Search Services. Informatica 38:11–19.

PÓCS, J. & . 2014. On the Inverse Problem for Generalized One-Sided Concept Lattices. Informatica 38:95–101.

SALAH, H. &, B. NACER, B. MAHMOUD. 2014. Semantic Annotations for Workflow Interoperability. Informatica 38:347–366.

ŠEVCECH, J. &, R. MŘO, M. HOLUB, M. BIELIKOVÁ. 2014. User Annotations as a Context for Related Document Search on the Web and Digital Libraries. Informatica 38:21–30.

SOMRAK, M. &, M. LUŠTREK, J. ŠUŠTERŠIČ, T. KRIVC, A. MLINAR, T. TRAVNIK, L. STEPAN, M. MAVSAR, M. GAMS. 2014. Tricorder: Consumer Medical Device for Discovering Common Medical Conditions. Informatica 38:81–88.

SZWED, P. &, P. SKRZYŃSKI, G. ROGUS, J. WEREWKA. 2014. SOAROAD: An Ontology of Architectural Decisions Supporting Assessment of Service Oriented Architectures. Informatica 38:31–42.

TOMAŠEV, N. & . 2014. The Role Of Hubness in High-dimensional Data Analysis. Informatica 38:387–388.

WANG, L. &, D. FU, L. WU. 2014. Incremental Hierarchical Fuzzy model generated from Multilevel Fuzzy Support Vector Regression Network. Informatica 38:367–375.

WONG, P. &, V. VARIKOTA, D. NGUYEN, A. ABUKMAIL. 2014. Automatic Android-based Wireless Mesh Networks. Informatica 38:313–320.

ZHOU, H. &, H. WANG. 2014. A Database-Based Two-Phase Algorithm For Efficient and Complete Detection of siRNA Off-Target Homology. Informatica 38:281–287.

ZHOU, X. &, K. LIU, Z. JIN, S. TIAN, Y. FU, L. QIN. 2014. Artificial Immune Based Cryptography Optimization Algorithm. Informatica 38:43–50.

ZUPANČIČ, D. &, B. CVETKOVIĆ. 2014. Smart-Home Energy Management in the Context of Occupants' Activity. Informatica 38:171–180.

JOŽEF STEFAN INSTITUTE

Jožef Stefan (1835-1893) was one of the most prominent physicists of the 19th century. Born to Slovene parents, he obtained his Ph.D. at Vienna University, where he was later Director of the Physics Institute, Vice-President of the Vienna Academy of Sciences and a member of several scientific institutions in Europe. Stefan explored many areas in hydrodynamics, optics, acoustics, electricity, magnetism and the kinetic theory of gases. Among other things, he originated the law that the total radiation from a black body is proportional to the 4th power of its absolute temperature, known as the Stefan–Boltzmann law.

The Jožef Stefan Institute (JSI) is the leading independent scientific research institution in Slovenia, covering a broad spectrum of fundamental and applied research in the fields of physics, chemistry and biochemistry, electronics and information science, nuclear science technology, energy research and environmental science.

The Jožef Stefan Institute (JSI) is a research organisation for pure and applied research in the natural sciences and technology. Both are closely interconnected in research departments composed of different task teams. Emphasis in basic research is given to the development and education of young scientists, while applied research and development serve for the transfer of advanced knowledge, contributing to the development of the national economy and society in general.

At present the Institute, with a total of about 900 staff, has 700 researchers, about 250 of whom are postgraduates, around 500 of whom have doctorates (Ph.D.), and around 200 of whom have permanent professorships or temporary teaching assignments at the Universities.

In view of its activities and status, the JSI plays the role of a national institute, complementing the role of the universities and bridging the gap between basic science and applications.

Research at the JSI includes the following major fields: physics; chemistry; electronics, informatics and computer sciences; biochemistry; ecology; reactor technology; applied mathematics. Most of the activities are more or less closely connected to information sciences, in particular computer sciences, artificial intelligence, language and speech technologies, computer-aided design, computer architectures, biocybernetics and robotics, computer automation and control, professional electronics, digital communications and networks, and applied mathematics.

The Institute is located in Ljubljana, the capital of the independent state of S**love**nia (or S♡nia). The capital today is considered a crossroad between East, West and Mediterranean Europe, offering excellent productive capabilities and solid business opportunities, with strong international connections. Ljubljana is connected to important centers such as Prague, Budapest, Vienna, Zagreb, Milan, Rome, Monaco, Nice, Bern and Munich, all within a radius of 600 km.

From the Jožef Stefan Institute, the Technology park "Ljubljana" has been proposed as part of the national strategy for technological development to foster synergies between research and industry, to promote joint ventures between university bodies, research institutes and innovative industry, to act as an incubator for high-tech initiatives and to accelerate the development cycle of innovative products.

Part of the Institute was reorganized into several high-tech units supported by and connected within the Technology park at the Jožef Stefan Institute, established as the beginning of a regional Technology park "Ljubljana". The project was developed at a particularly historical moment, characterized by the process of state reorganisation, privatisation and private initiative. The national Technology Park is a shareholding company hosting an independent venture-capital institution.

The promoters and operational entities of the project are the Republic of Slovenia, Ministry of Higher Education, Science and Technology and the Jožef Stefan Institute. The framework of the operation also includes the University of Ljubljana, the National Institute of Chemistry, the Institute for Electronics and Vacuum Technology and the Institute for Materials and Construction Research among others. In addition, the project is supported by the Ministry of the Economy, the National Chamber of Economy and the City of Ljubljana.

Jožef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
Tel.:+386 1 4773 900, Fax.:+386 1 251 93 85
WWW: http://www.ijs.si
E-mail: matjaz.gams@ijs.si
Public relations: Polona Strnad

INFORMATICA

AN INTERNATIONAL JOURNAL OF COMPUTING AND INFORMATICS

INVITATION, COOPERATION

Submissions and Refereeing

Please submit a manuscript to: http://www.informatica.si/Editors/PaperUpload.asp. At least two referees outside the author's country will examine it, and they are invited to make as many remarks as possible from typing errors to global philosophical disagreements. The chosen editor will send the author the obtained reviews. If the paper is accepted, the editor will also send an email to the managing editor. The executive board will inform the author that the paper has been accepted, and the author will send the paper to the managing editor. The paper will be published within one year of receipt of email with the text in Informatica MS Word format or Informatica LaTeX format and figures in .eps format. Style and examples of papers can be obtained from http://www.informatica.si. Opinions, news, calls for conferences, calls for papers, etc. should be sent directly to the managing editor.

QUESTIONNAIRE

☐ Send Informatica free of charge

☐ Yes, we subscribe

Please, complete the order form and send it to Dr. Drago Torkar, Informatica, Institut Jožef Stefan, Jamova 39, 1000 Ljubljana, Slovenia. E-mail: drago.torkar@ijs.si

Since 1977, Informatica has been a major Slovenian scientific journal of computing and informatics, including telecommunications, automation and other related areas. In its 16th year (more than twenty years ago) it became truly international, although it still remains connected to Central Europe. The basic aim of Informatica is to impose intellectual values (science, engineering) in a distributed organisation.

Informatica is a journal primarily covering intelligent systems in the European computer science, informatics and cognitive community; scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the Refereeing Board.

Informatica is free of charge for major scientific, educational and governmental institutions. Others should subscribe (see the last page of Informatica).

ORDER FORM – INFORMATICA

Name: .....

Title and Profession (optional): .....

.....

Home Address and Telephone (optional): .....

.....

Office Address and Telephone (optional): .....

.....

E-mail Address (optional): .....

Signature and Date: .....

Informatica WWW:

http://www.informatica.si/

Referees from 2008 on:

A. Abraham, S. Abraham, R. Accornero, A. Adhikari, R. Ahmad, G. Alvarez, N. Anciaux, R. Arora, I. Awan, J. Azimi, C. Badica, Z. Balogh, S. Banerjee, G. Barbier, A. Baruzzo, B. Batagelj, T. Beaubouef, N. Beaulieu, M. ter Beek, P. Bellavista, K. Bilal, S. Bishop, J. Bodlaj, M. Bohanec, D. Bolme, Z. Bonikowski, B. Bošković, M. Botta, P. Brazdil, J. Brest, J. Brichau, A. Brodnik, D. Brown, I. Bruha, M. Bruynooghe, W. Buntine, D.D. Burdescu, J. Buys, X. Cai, Y. Cai, J.C. Cano, T. Cao, J.-V. Capella-Hernández, N. Carver, M. Cavazza, R. Ceylan, A. Chebotko, I. Chekalov, J. Chen, L.-M. Cheng, G. Chiola, Y.-C. Chiou, I. Chorbev, S.R. Choudhary, S.S.M. Chow, K.R. Chowdhury, V. Christlein, W. Chu, L. Chung, M. Ciglarič, J.-N. Colin, V. Cortellessa, J. Cui, P. Cui, Z. Cui, D. Cutting, A. Cuzzocrea, V. Cvjetkovic, J. Cypryjanski, L. Čehovin, D. Čerepnalkoski, I. Čosić, G. Daniele, G. Danoy, M. Dash, S. Datt, A. Datta, M.-Y. Day, F. Debili, C.J. Debono, J. Dedič, P. Degano, A. Dekdouk, H. Demirel, B. Demoen, S. Dendamrongvit, T. Deng, A. Derezinska, J. Dezert, G. Dias, I. Dimitrovski, S. Dobrišek, Q. Dou, J. Doumen, E. Dovgan, B. Dragovich, D. Drajic, O. Drbohlav, M. Drole, J. Dujmović, O. Ebers, J. Eder, S. Elaluf-Calderwood, E. Engström, U. riza Erturk, A. Farago, C. Fei, L. Feng, Y.X. Feng, B. Filipič, I. Fister, I. Fister Jr., D. Fišer, A. Flores, V.A. Fomichov, S. Forli, A. Freitas, J. Fridrich, S. Friedman, C. Fu, X. Fu, T. Fujimoto, G. Fung, S. Gabrielli, D. Galindo, A. Gambarara, M. Gams, M. Ganzha, J. Garbajosa, R. Gennari, G. Georgeson, N. Gligorić, S. Goel, G.H. Gonnet, D.S. Goodsell, S. Gordillo, J. Gore, M. Grčar, M. Grgurović, D. Grosse, Z.-H. Guan, D. Gubiani, M. Guid, C. Guo, B. Gupta, M. Gusev, M. Hahsler, Z. Haiping, A. Hameed, C. Hamzaçebi, Q.-L. Han, H. Hanping, T. Härder, J.N. Hatzopoulos, S. Hazelhurst, K. Hempstalk, J.M.G. Hidalgo, J. Hodgson, M. Holbl, M.P. Hong, G. Howells, M. Hu, J. Hyvärinen, D. Ienco, B. Ionescu, R. Irfan, N. Jaisankar, D. Jakobović, K. Jassem, I. Jawhar, Y. Jia, T. Jin, I. Jureta, Đ. Juričić, S. K, S. Kalajdziski, Y. Kalantidis, B. Kaluža, D. Kanellopoulos, R. Kapoor, D. Karapetyan, A. Kassler, D.S. Katz, A. Kaveh, S.U. Khan, M. Khattak, V. Khomenko, E.S. Khorasani, I. Kitanovski, D. Kocev, J. Kocijan, J. Kollár, A. Kontostathis, P. Korošec, A. Koschmider, D. Košir, J. Kovač, A. Krajnc, M. Krevs, J. Krogstie, P. Krsek, M. Kubat, M. Kukar, A. Kulis, A.P.S. Kumar, H. Kwaśnicka, W.K. Lai, C.-S. Laih, K.-Y. Lam, N. Landwehr, J. Lanir, A. Lavrov, M. Layouni, G. Leban, A. Lee, Y.-C. Lee, U. Legat, A. Leonardis, G. Li, G.-Z. Li, J. Li, X. Li, X. Li, Y. Li, Y. Li, S. Lian, L. Liao, C. Lim, J.-C. Lin, H. Liu, J. Liu, P. Liu, X. Liu, X. Liu, F. Logist, S. Loskovska, H. Lu, Z. Lu, X. Luo, M. Luštrek, I.V. Lyustig, S.A. Madani, M. Mahoney, S.U.R. Malik, Y. Marinakis, D. Marinčič, J. Marques-Silva, A. Martin, D. Marwede, M. Matijašević, T. Matsui, L. McMillan, A. McPherson, A. McPherson, Z. Meng, M.C. Mihaescu, V. Milea, N. Min-Allah, E. Minisci, V. Mišić, A.-H. Mogos, P. Mohapatra, D.D. Monica, A. Montanari, A. Moroni, J. Mosegaard, M. Moškon, L. de M. Mourelle, H. Moustafa, M. Možina, M. Mrak, Y. Mu, J. Mula, D. Nagamalai, M. Di Natale, A. Navarra, P. Navrat, N. Nedjah, R. Nejabati, W. Ng, Z. Ni, E.S. Nielsen, O. Nouali, F. Novak, B. Novikov, P. Nurmi, D. Obrul, B. Oliboni, X. Pan, M. Pančur, W. Pang, G. Papa, M. Paprzycki, M. Paralič, B.-K. Park, P. Patel, T.B. Pedersen, Z. Peng, R.G. Pensa, J. Perš, D. Petcu, B. Petelin, M. Petkovšek, D. Pevec, M. Pičulin, R. Piltaver, E. Pirogova, V. Podpečan, M. Polo, V. Pomponiu, E. Popescu, D. Poshyvanyk, B. Potočnik, R.J. Povinelli, S.R.M. Prasanna, K. Pripužić, G. Puppis, H. Qian, Y. Qian, L. Qiao, C. Qin, J. Que, J.-J. Quisquater, C. Rafe, S. Rahimi, V. Rajkovič, D. Raković, J. Ramaekers, J. Ramon, R. Ravnik, Y. Reddy, W. Reimche, H. Rezankova, D. Rispoli, B. Ristevski, B. Robič, J.A. Rodriguez-Aguilar, P. Rohatgi, W. Rossak, I. Rožanc, J. Rupnik, S.B. Sadkhan, K. Saeed, M. Saeki, K.S.M. Sahari, C. Sakharwade, E. Sakkopoulos, P. Sala, M.H. Samadzadeh, J.S. Sandhu, P. Scaglioso, V. Schau, W. Schempp, J. Seberry, A. Senanayake, M. Senobari, T.C. Seong, S. Shamala, c. shi, Z. Shi, L. Shiguo, N. Shilov, Z.-E.H. Slimane, F. Smith, H. Sneed, P. Sokolowski, T. Song, A. Soppera, A. Sorniotti, M. Stajdohar, L. Stanescu, D. Strnad, X. Sun, L. Šajn, R. Šenkeřík, M.R. Šikonja, J. Šilc, I. Škrjanc, T. Štajner, B. Šter, V. Štruc, H. Takizawa, C. Talcott, N. Tomasev, D. Torkar, S. Torrente, M. Trampuš, C. Tranoris, K. Trojacanec, M. Tschierschke, F. De Turck, J. Twycross, N. Tziritas, W. Vanhoof, P. Vateekul, L.A. Vese, A. Visconti, B. Vlaovič, V. Vojisavljević, M. Vozalis, P. Vračar, V. Vranić, C.-H. Wang, H. Wang, H. Wang, H. Wang, S. Wang, X.-F. Wang, X. Wang, Y. Wang, A. Wasilewska, S. Wenzel, V. Wickramasinghe, J. Wong, S. Wrobel, K. Wrona, B. Wu, L. Xiang, Y. Xiang, D. Xiao, F. Xie, L. Xie, Z. Xing, H. Yang, X. Yang, N.Y. Yen, C. Yong-Sheng, J.J. You, G. Yu, X. Zabulis, A. Zainal, A. Zamuda, M. Zand, Z. Zhang, Z. Zhao, D. Zheng, J. Zheng, X. Zheng, Z.-H. Zhou, F. Zhuang, A. Zimmermann, M.J. Zuo, B. Zupan, M. Zuqiang, B. Žalik, J. Žižka,

Informatica

An International Journal of Computing and Informatics

Web edition of Informatica may be accessed at: http://www.informatica.si.

Subscription Information Informatica (ISSN 0350-5596) is published four times a year in Spring, Summer, Autumn, and Winter (4 issues per year) by the Slovene Society Informatika, Litostrojska cesta 54, 1000 Ljubljana, Slovenia.
The subscription rate for 2014 (Volume 38) is
– 60 EUR for institutions,
– 30 EUR for individuals, and
– 15 EUR for students
Claims for missing issues will be honored free of charge within six months after the publication date of the issue.

Typesetting: Borut Žnidar.
Printing: ABO grafika d.o.o., Ob železnici 16, 1000 Ljubljana.

Orders may be placed by email (drago.torkar@ijs.si), telephone (+386 1 477 3900) or fax (+386 1 251 93 85). The payment should be made to our bank account no.: 02083-0013014662 at NLB d.d., 1520 Ljubljana, Trg republike 2, Slovenija, IBAN no.: SI56020830013014662, SWIFT Code: LJBASI2X.

Informatica is published by Slovene Society Informatika (president Niko Schlamberger) in cooperation with the following societies (and contact persons):
Robotics Society of Slovenia (Jadran Lenarčič)
Slovene Society for Pattern Recognition (Janez Perš)
Slovenian Artificial Intelligence Society (Dunja Mladenić)
Cognitive Science Society (Urban Kordeš)
Slovenian Society of Mathematicians, Physicists and Astronomers (Andrej Likar)
Automatic Control Society of Slovenia (Sašo Blažič)
Slovenian Association of Technical and Natural Sciences / Engineering Academy of Slovenia (Vojteh Leskovšek)
ACM Slovenia (Andrej Brodnik)

Informatica is financially supported by the Slovenian research agency from the Call for co-financing of scientific periodical publications.

Informatica is surveyed by: ACM Digital Library, Citeseer, COBISS, Compendex, Computer & Information Systems Abstracts, Computer Database, Computer Science Index, Current Mathematical Publications, DBLP Computer Science Bibliography, Directory of Open Access Journals, InfoTrac OneFile, Inspec, Linguistic and Language Behaviour Abstracts, Mathematical Reviews, MatSciNet, MatSci on SilverPlatter, Scopus, Zentralblatt Math

Volume 38 Number 4 December 2014 **ISSN 0350-5596**

Informatica

An International Journal of Computing and Informatics

Informatica **38** (2014) Number 4, pp. 311–391