University of Bath

Modelling, Monitoring and Evaluation to Support Automatic Engineering Process Management

Lei Shi[1], Linda Newnes[1], Steve Culley[1] and Bruce Allen[2]

Abstract

Process management is considered as an essential approach to improve the performance of an enterprise. The process of an engineering project is considered to be a formalised workflow accompanied by a set of decisions. With decisions being made by taking account of information from various sources, the operation and management of modern engineering projects has to deal with increasing amounts of dynamic and changing project information. Understanding and interpreting this information for use in process management can generate challenges in practice. This might be caused by constraints of time and resource, the distributed structure of the information and a lack of modelled domain knowledge. To address these challenges, the research described in this paper focuses on techniques which support the automation of the process management of engineering projects, from a data-driven perspective. The research includes elements of process modelling, monitoring and evaluation of such projects, through a proposed automatic process analysis system. The proposed system works with live and historical data. Within this paper, the design and implementation of the system is described. The use of techniques such as autonomic computing, data mining and knowledge management technologies are shown, and the system functionality is demonstrated through the use of a dataset from an aerospace organisation.

Introduction

The process of an engineering project should be a formalised workflow accompanied by a set of decisions, which are made by taking into account various pieces of information, such as project objectives, required resources and the dynamics of the working environment. Process management has been considered as an essential approach to improving the performance of an enterprise (Haddar *et al.* 2014). In recent research, various process management models have been proposed covering the fields of process modelling, process reuse, complexity identification, process standardisation and process optimisation (Schäfermeyer *et al.* 2012; Shen *et al.* 2006).

However it has become clear that the operation and management of modern engineering projects need to deal with increasing volumes of dynamic information (Eppler & Mengis 2004). The related actions may involve receiving, understanding and interacting with this information via various project partners. It will also require the creation of solutions to interpret, articulate, clarify and make decisions based on this the dynamic information. Hence, understanding this dynamic project information and reusing the information for process management purposes generates challenges in practice. The challenges can be caused by

- Constraints of time and resource;
- Distributed structure of information;
- Lack of modelled domain knowledge.

The details of these challenges are discussed below.

Constraints of time and resource: As manufacturing in the globalised environment faces intensive competition, the need to maintain a high level of profitability, reduce operation/management cost and improve time/resource efficiency become critical requirements for engineering companies (Al-Najjar & Alsyouf 2004; Airbus 2002). To support the related decision making tasks, the essential information generated at each project stage needs to be captured and assessed collectively. The detailed analysis of this information on a detailed level could enable decision makers to have a comprehensive understanding about project process evolutions and characteristic changes, such as the interactions between project actors, the dependency between project components, and the performance changes regarding different project stages.

However, such information typically is considerable, especially from large-scale projects or projects with complex processes. Thus, due to the conflict between the amount of information and the limitation of time/human resource, human-centered decision-making struggles to fully take

[1] Department of Mechanical Engineering,
University of Bath, Bath BA2 7AY, UK.
[2] AIRBUS Operations Ltd,
Filton, Bristol BS99 7AR, UK.

Corresponding author:
Lei Shi, Department of Mechanical Engineering,
University of Bath, Bath BA2 7AY, UK.
Email: l.shi@bath.ac.uk

account of all the necessary information. This also may mean that the decision-making and related problem solving tasks, may have an outcome bias in areas where the project actor has specific knowledge and expertise, rather than the an overall project perspective.

Distributed structure of information: From an organisational perspective, the globalisation of product design, manufacturing and services enables the partners of collaborative engineering projects to be located in various regions/countries (Nieusma & Riley 2010; Abdalla 1999). As a result, the operation and management of such engineering projects are critically dependent upon distributed communications, together with large amounts of digitalised documents and dynamically generated workflows (Carey *et al.* 2013; Xie *et al.* 2011). In practice, the management of distributed data is still not straight forward or well supported. There are a number of unsolved issues in system design, data storage, data integration and transaction management (Agrawal *et al.* 2010; Deelman & Chervenak 2008). Therefore, the distributed project information, together with the decentralised project structure, could significantly increase the complexity of the project process and the difficulty of the project operation/management. This could prevent project actors from having awareness of potential problems throughout the project lifecycle.

Lack of modelled domain knowledge: The shortage of experienced labour has become a serious issue in various industrial sectors, including aerospace, automotive and other engineering sectors (Dychtwald *et al.* 2013; Lewin *et al.* 2009; Weber *et al.* 2007). The high rate of retirement and turnover of experienced staff can fundamentally change the structure of established departments, decrease their average expertise level and operational efficiency. To avoid these negative impacts, knowledge management (KM) has been suggested as a solution (Lindner & Wald 2011; Rosemann & vom Brocke 2015). However, using top-down methods to model the knowledge in large organisations, even in small-to medium-sized enterprises (SMEs) remains a challenge. This is often due to the KM process itself being complicated and time/resource consuming (Hislop 2013; Hörisch *et al.* 2014).

According to recent research, solving these three key challenges are considered to be critical for improving the capability of information management, knowledge management, and workflow management for collaborative engineering projects (Weber *et al.* 2007; Xie *et al.* 2011; Scheer & Nüttgens 2000). It is these issues that drive the research work described in this paper.

The purpose of the paper

In order to automate the process management of an engineering project, the following requirements are considered in this paper,

- Modelling the operational process from the project data and defining process normality;
- In real time monitor the process and ascertain the level of process normality;
- Utilisation of the captured/modelled domain knowledge to facilitate process management and its related analytical tasks.

Based on these requirements, a system is proposed that integrates three main functionalities process modelling, process monitoring and process evaluation. The design and implementation of the system and its related analytical approaches apply autonomic computing, data mining and knowledge management technologies.

To demonstrate the creation and the evaluation of the proposed system, case studies from the Aerospace service sector (Departments which provide in-service support) are shown. In-Service departments for aircraft play a vital and increasingly important role in delivering services such as modifications and upgrades, maintenance and emergency repairs to airline operators of all types. In addition, they have the opportunity to collect feedback from the stakeholders e.g., airline operators, contractors and specialist suppliers. The information contained in the feedback can then be used to improve aircraft design in the future.

As is typical of many collaborative engineering projects, In-Service projects contain complex processes with asynchronous and synchronous collaborations (Vianello *et al.* 2010). Their execution processes need to deal with various constraints in terms of time, budget and resource. In the manufacturing environment, the process management of In-Service projects still relies on human-centered decision-making approaches. This decision-making requires the understanding, modelling and representing of large amount of project information. Consequently, the monitoring and evaluation of multiple processes on a real-time basis is a challenge for the In-Service departments. This challenge becomes common issue in the process management of engineering projects today. To demonstrate that the proposed automatic process analysis system (APAS) can address such a challenge, the research presented in this paper utilises information and knowledge from 396 In-Service case studies, to research the design, implementation and generalisation of the system and the underpinning techniques utilised.

The remainder of this paper is organised as follows. Section 2 reviews the related work in the fields of process management, autonomic computing and data mining in manufacturing. Section 3 introduces the proposed automatic process analysis system, and the technical details of the proposed approaches to be used in APAS, are described in Section 4. The experimental results are evaluated in Section 5 and Section 6 provides the conclusions from the analysis.

Related Work

Process Management

Process management aims to improve the efficiency and effectiveness of organising project activities, together with facilitating the understanding of inter relationships among the activities (Weske 2012). As an extension of workflow management, it involves various information technologies, specific knowledge and associated data, to support the design, enactment, management and analysis of operational processes (de Medeiros *et al.* 2008; Van Der Aalst 2013).

According to recent research, various process management models have been developed and applied in different sectors to solve their practical problems. Benner & Tushman (2002) revealed that the activities of process management have associations with the increase of innovations and the

share of innovations. This demonstrates that process management can be used to enhance the technical innovations, and broaden the existing knowledge of organisations. Pino *et al.* (2008) indicated that the use of process management in software engineering projects is an effective way to improve the project management, documentation management, requirements change management, process establishment, configuration management and requirements elicitation. de Mast *et al.* (2011) proposed a conceptual framework of process management for healthcare projects, by considering the micro processes, tasks and resources from the project workflow, which is used to improve the process efficiency, resource management and organisational performances. Rao *et al.* (2012) introduced a model that integrates ontologies and knowledge maps with process re-engineering approaches, and it is used to improve the efficiency of business processes. Lerner *et al.* (2010) proposed an approach to handling exception patterns of modelled process, which could capture the relationship between exception handling tasks and the normative process.

Due to the dynamics in manufacturing environments, the definition and technical specification of process management are difficult to be consolidated and formalised. As a result, the creation or selection of suitable process management models remains a challenge (Ko 2009). It is also challenging to apply the existing process management models within manufacturing processes. At times the application of such model is considered less important by project actors, e.g., a model could be applied at the early stage of a project, and then intentionally or unintentionally ignored at the following stages (Michael Gnatz & Rausch 2004). To improve the usability of process management models, researchers have shown it is necessary to reduce the human intervention. Hence, computer aided technologies need to be integrated. In the automotive industry, ICT based support has been used to support the process management and support the process development (Müller *et al.* 2006). The technologies have also been used to support process planning in collaborative manufacturing and product lifecycle management (Ming *et al.* 2008).

A number of gaps become clear from this review, namely that there are only piecemeal attempts at solutions, that it is important to reduce human intervention and that the time-based aspects has been largely overlooked.

Autonomic Computing

Information systems in modern manufacturing have increasingly sophisticated structures, as they need to include more components, deal with distributed resources, process heterogeneous data, and support multiple users with varying levels of access. These factors make the development, configuration and management of such systems to become more difficult. Researchers from IBM first revealed that the operational and managerial mechanisms of complex information systems have certain degrees of similarity with the human biological system (Kephart & Chess 2003). For example, the biological system such as autonomic nervous system, can manage essential body functions autonomously, such as the monitoring of heartbeat, maintenance of blood sugar levels and body temperature, without any effort from the human. This type of self-management features are considered particularly useful for improving the autonomy of complex information systems (Sterritt *et al.* 2005).

On the basis of this concept, *autonomic computing* is designed as an automatic approach that simulates the autonomic nervous system. This could be used to integrate self-management functionalities into information systems, i.e., to control the function of computation and system operation related tasks without human intervention.

Autonomic computing systems are typically referred to as any information system that involves autonomic computing. In its self-management process, the system needs to sense the temporal status of each internal component and the condition changes of the external environment. It then takes appropriate actions based on the sensed information. The control loop in autonomic computing system covers the aspects of *self-configuration*, *self-optimisation*, *self-healing* and *self-protection* (Kephart & Chess 2003). In general, the *self-configuration* means the system can automatically adjust itself according to the pre-designed high-level guidance; the *self-optimisation* means the system can constantly monitor and adjust itself during the operation, continually improve its performance and efficiency; the *self-healing* means the system has the capability of handling the failures caused by itself; and the *self-protection* means the system could automatically anticipate and prevent attacks or failures caused by external sources (IBM 2006).

To implement complex tasks, the autonomic computing system integrates various types of components, including software, hardware, services or combination of them. Each system component is treated as a single *autonomic element* that is designed to automatically perform certain behaviours. The autonomic element contains a communication mechanism that enables it to communicate with other elements in the system. Based on different component combinations, the system could have the capability to perform tasks in different manufacturing environments. For a particular environment, the key behaviours of the autonomic elements need to be pre-defined according to the environment condition and user requirements, and then organised into hierarchical structures. The functionality of the system is therefore dependent upon the internal behaviours of each autonomic element, and the relationships between the autonomic elements (Huebscher & McCann 2008; Sterritt *et al.* 2005).

Recently, various types of autonomic computing systems have been proposed and applied in different fields. Kim *et al.* (2011) proposed an autonomic model to manage the application workflow in hybrid computing infrastructure. Here the system and application states are minitored and then the applications and resources are adapted to respond to the changes of environment. Caton & Rana (2012) stated that large-scale and multi-users information management systems, such as cloud services, could utilise autonomic management to improve their reliability and the predictability of resource management. Fallon & O'Sullivan (2013) integrated semantic technologies into an autonomic model, and then applied the model to manage end-user service quality. These ideas have been used in the creation of the approaches used by the research reported in this paper.

Data Mining in Manufacturing

In modern manufacturing, more and more information management systems have started to integrate advanced analytical approaches from data mining and machine learning fields, in order to address the issues on information understanding and reusing, and to fulfil the information needs about process management and evaluation. Data mining is an important tool to support the daily information management tasks and discover the knowledge from manufacturing databases (Choudhary *et al.* 2009). It is also considered as the fundamental tool for developing more advanced information management systems. For example, it can be used to achieve the functions including predictive maintenance, fault detection, quality control and customer relationship management (Harding *et al.* 2006; Köksal *et al.* 2011). Under the scope of data mining, various analytical technologies have been proposed, such as classification, clustering, pattern identification/extraction, feature selection/modelling and visualisation. By using these approaches, the raw data from engineering projects can be automatically organised and analysed, enabling project actors to gain a more comprehensive understandings of project characteristics without excessive effort (Wang 2007).

Recent research demonstrated that data mining is playing an important role in supporting information management in manufacturing. Gröger *et al.* (2012) revealed that data mining can be used to optimise the workflow-based business process and generate decision rules/trees for process analysis use. Shi *et al.* (2014a) proposed an approach integrating data mining with domain knowledge, which automatically identifies the temporal changes of an engineering project process from workflow related data. Shi *et al.* (2014b) also revealed that the patterns contained in the project documentations are particularly useful for the planning of similar projects. Meanwhile, the identified patterns can be directly transferred into a knowledge base for future reuse.

Data generation of an engineering project is accompanied with the project operation process; hence the information contained in project data can be used to reconstruct the actual project process. Data mining has been considered to have the capability to discover the meaningful patterns from the data, and also to identify the relations between the patterns.

On this basis, data mining has been selected as suitable to be used as the *'virtual sensor'* and *'information provider'* in the autonomic process management system, which could automatically identify and extract process-related information from project data.

Automatic Process Analysis System (APAS)

In modern manufacturing, collaborative engineering projects are concurrent, heterogeneous, and constrained. In general, 'concurrent' implies that, to fit the demands of changing markets and consumers, the project teams need to simultaneously implement multiple tasks/projects on different product/service lines; 'heterogeneous' implies that projects could require different processes due to their different types, priority settings and technical requirements; 'constrained' implies that the projects are performed under constraints, e.g., the time pressures, the limitation of humans or other types of resources.

To automate the process management for collaborative engineering projects, two research challenges are identified,

- How to improve the manageability and understand-ability of project information;
- How to reduce human intervention in data analysis and decision-making related tasks.

To address these challenges, the proposed automatic process analysis system (APAS) integrates data mining technologies, natural language processing and knowledge bases. *Knowledge bases* are used as the high level guidances to perform analytical tasks. Project actors, e.g., engineers and managers, are able to add new knowledge to the knowledge bases at any time, or to edit the existing concepts in the knowledge bases if needed. These behaviours could integrate the dynamics of environments into the knowledge bases.

The system encompasses three main phases, i.e., *the modelling phase*, *the monitoring phase* and *the evaluation phase* (see Figure 1). These phases are executed in a predefined order, each having specific functionalities. For example, *the modelling phase* focuses on processing the captured project data, identifying process features from the data and selecting additional features from modelled knowledge bases; *the monitoring phase* categorises modelled processes by using clustering/classification methods, segment such processes based on predefined time intervals and compares the similarity between ongoing processes and segmented processes; *the evaluation phase* measures the process normality, and generates interactive visualisations for modelled processes.

The components included in these phases and the technical details are introduced as follows.

Modelling Phase: as shown in Figure 2, the components involved in this phase include *data processing*, *knowledge mapping*, *feature modelling* and *process modelling*. At the initial stage, the distributed project data is captured and integrated from different sources, including those related to communication, workflow and environment. The integrated data is treated as the system input, and then processed by the analytical modules, including *meta-data analysis*, *content analysis* and *semantic analysis*. For the analysis, certain types of knowledge bases, e.g., lexical ontologies, are applied to support the *semantic analysis* module. This module is used to identify and extract semantic features, e.g., important terminologies or phrases, from content-related data such as project descriptions, objectives or related communications. In the *knowledge mapping* module, the extracted information is used to select domain specific knowledge bases to facilitate further analysis, such as *feature modelling* and *process modelling*. The *feature identification* module aims to identify process-related features from the data, e.g., activity names, named entities, document names, department information, client information, etc. In the *feature selection* module, the identified process-related features are filtered/weighted based on the selected domain specific knowledge bases, and then the most important features are passed to the next step, activity mapping. In the *activity mapping* module, each identified activity is treated as a high-level indication of process-related features, which can be used to map/link related features together. In the end, the *sequencing* module converts the features with their timestamps into feature-based sequences (modelled process).
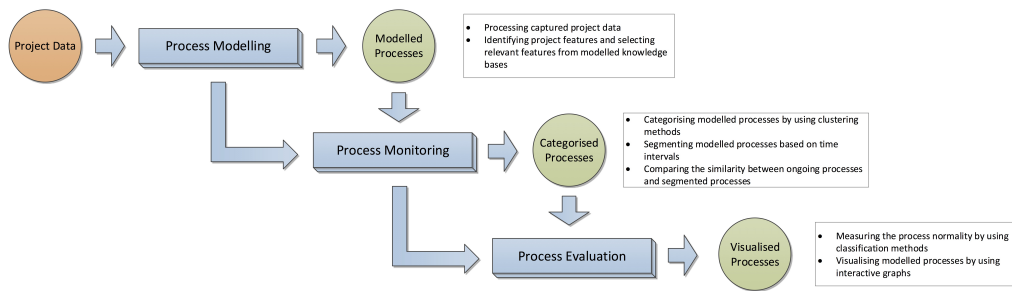
Figure 1. The main phases of the system

Monitoring Phase: as shown in Figure 3, the components involved in this phase include *process visualisation* and *process monitoring*. In this phase, an interactive interface for process visualisation is provided to the project actors, enabling them to browse multiple modelled processes simultaneously. *Process monitoring* contains two modules *clustering* and *classification*, which are used to categorise the modelled processes into groups. The *clustering* module applies the unsupervised approach to categorise the modelled processes into groups by measuring the similarities/dissimilarities between the processes. In other words, the processes in the same group need to have a certain level of similarity with each other. This module aims to assist the project actors through quick access to projects with similar processes. The *classification* module applies the supervised approach, by using a training set, to discriminate the category a process belongs to. The training set contains a collection of labelled processes, and each label corresponds to a pre-defined category. The processes together with their labels from the training set are used to train the classifier in how to categorise unlabelled processes. During the classification, a specific label will be assigned to an unlabelled process, if the process has certain level of similarity with the labelled processes. With this module, when a new project takes place, it will be automatically categorised into the appropriate category. This module aims to help the project actors effectively compare the new/current project process with other similar projects.

Evaluation Phase: as shown in Figure 4, the components involved in this phase include *process modelling*, *content retrieval*, *process segmentation*, *normality measure* and *process visualisation*. In the initial stage, the data of ongoing projects is processed by the *process modelling* module. The *content retrieval* module then automatically retrieves similar projects from the dataset based on the identified process features. Based on the elapsed time of ongoing projects, the processes of retrieved projects are segmented by the *process segmentation* module. Next, the process normality of ongoing projects is measured by considering the sequence similarity between the ongoing project process and retrieved project processes. The outputs from the analysis generated by the *normality measure* module, enable project actors to ascertain whether the ongoing project is not behaving in a normal fashion.

In APAS, the analytical modules are designed to be automated. To achieve their functionalities, data mining and machine learning based approaches are required. The technical details of these approaches are introduced in the following section.

Automatic Approaches on Process Management

To deal with different projects, a critical requirement is to handle their heterogeneous data in a uniform way, ensuring the information contained in such data can be represented consistently. It then enables both computers and human to use the same approach for understanding and evaluating the process structures and project characteristics.

In APAS, a hybrid representation is proposed and used to fulfil the requirement. The core consists of feature based vectors and sequences, which are created by considering two types of information, i.e., semantic and process-related. Accordingly, five analytical approaches are used to achieve the automatic creation of these vectors and sequences. These approaches include feature identification, feature selection, process sequencing, process similarity measure and a process normality measure.

Feature Identification and Selection

Using p_i to denote a project, $p_i = \{M, C\}$ denote its related dataset, where M is the set of metadata, and C is the set of content information. The metadata contains the bottom-level description and attributes related information of the project data itself, e.g., the file version, creator, issued by, modified date, storage path, etc. The content information contains descriptive information, semantic information and knowledge related information about the project, e.g., the defined objectives, encountered problems, applied solutions, contained communications, issued documentations, involved technical details, etc.

For a general knowledge base K_G, contains terminologies, entities and general concepts related to the common characteristics of projects. Let F_G denote a set of semantic features in K_G, and $f_{G,m}$ denote a single feature, where $f_{G,m} \in F_G$ and $F_G = (K_G \cap M) \cup (K_G \cap C)$. The feature vector of project p_i is,

$$v_i = [f_{G,1}, f_{G,2}, ..., f_{G,m}]^T \qquad (1)$$

where m is the total number of semantic features identified from the project data.

For a process-related knowledge base K_P, contains process-related concepts, activities, document types, and their related dependencies and weightings. Let F_P denote
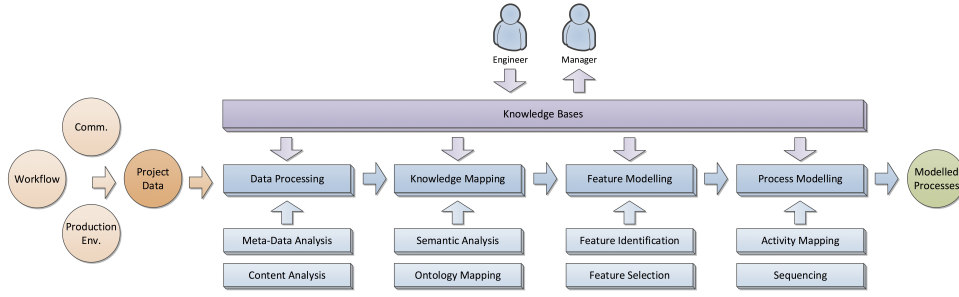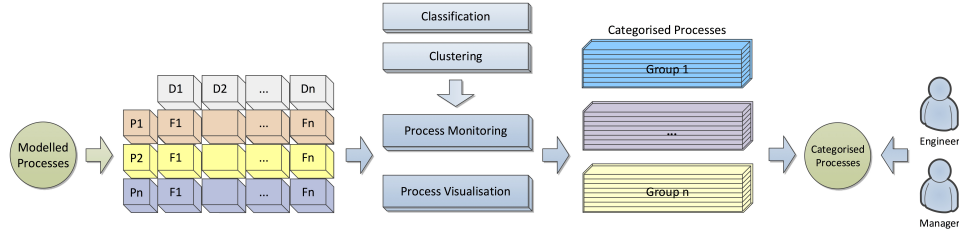
Figure 2. Modelling phase

Figure 3. Monitoring phase
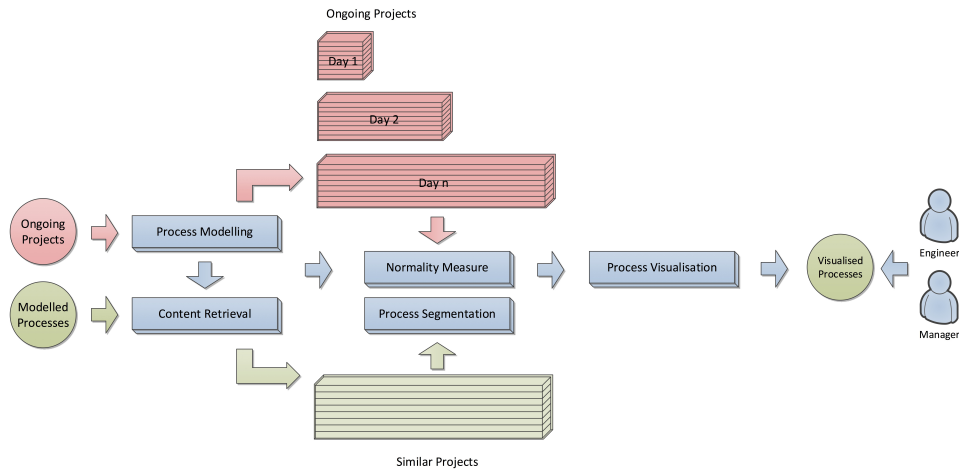
Figure 4. Evaluation phase

a set of process-related features in K_P, and $f_{P,n}$ denote a single feature, where $f_{P,n} \in F_P$ and $F_P = K_P \cap C$. If a process sequence containing multiple features, the index of a single feature is determined by the timestamp of the feature. For example, $f_{P,i}$ and $f_{P,j}$ are the features in one sequence, $ts_{P,i}$ and $ts_{P,j}$ denote their timestamps respectively, and the indexes of these features satisfy $i < j$, if the timestamps of them satisfy $ts_{P,i} < ts_{P,j}$. According to these, the process sequence of the project p_i is represented as,

$$s_i = \sum_{n=1}^{|F_P|} w_n \cdot f_{P,n} \qquad (2)$$

where n is the index of feature $f_{P,n}$; $|F_P|$ is the size of feature set F_P; and w_n is a weighting function, $w_n \in [0, 1]$.

Time-based Sequence/Vector Segmentation

In practice, a project process is usually dynamic over time, thus adding a time dimension onto the data representation is necessary. This time dimension will enable the investigation of process evolution for both ongoing (i.e., uncompleted) and completed projects, as well as the comparison of process similarity between them. For this purpose, time-based sequence/vector segmentation are proposed and applied to convert a modelled process into multiple sub-processes based on time intervals.

For a given timestamp ts, it can be used to indicate the absolute time, e.g., x-th hour/day/step, or the relative time, e.g., $x\%$ of the project timeline T. Let F_P' denote a process-related feature set with ts, for its contained features, their timestamps need to satisfy $ts_{P,n} < ts$. According to this, the

sub-sequence of project p_i is represented as,

$$s'_i = \sum_{n=1}^{|F'_P|} (w_n \cdot f_{P,n})_{ts}, ts \in \tau$$

$$\tau = \{\sum_{j=1}^{|seg|} (j \cdot \frac{T}{|seg|})\}, j \in [1, |seg|]$$

(3)

where τ is the set of time intervals regarding the timeline, and $|seg|$ is the number of segmentations of the timeline.

Similarly, using F'_G denote a general feature set with ts, for its contained features, the timestamps need to satisfy $ts_{G,n} < ts$. According to this, the sub-vector of project p_i is represented as,

$$v'_i = [f_{G,1}, f_{G,2}, ..., f_{G,|F'_G|}]^T$$

(4)

where $|F'_G|$ is the size of feature set F'_G.

In this segmentation process, F'_P and F'_G are the truncated sets being converted from the original sets F_P and F_G. The contained features in these truncated sets are determined/adjusted by the setting of ts. For example, if ts is the last date of T, then $F'_P = F_P$ and $F'_G = F_G$, which means the segmented sequence and vector are equal to the original ones; if ts is equal to 10% of the timeline T, F'_P and F'_G will only contain the features which are generated within the 10% of project progress.

Process Similarity Measure

The similarity measure of proposed data representation considers semantic similarity and sequence similarity. In the monitoring phase, project processes with certain levels of combined similarity are categorised into the same group. The similarity measure process is described as: giving a threshold $\epsilon \in (0, 1)$, the processes of project p_i and p_j are considered as similar, if and only if $sim_{sem}(p_i, p_j) + sim_{seq}(p_i, p_j) \geq \epsilon$.

To measure the semantic similarity, the *Vector Space Model* (VSM) is applied. For a set of projects, all of the contained semantic features from them are used to construct a feature space. The similarity between any two projects is then measured based on the angle between their related feature vectors in the feature space. For the given feature vectors v_i and v_j, the similarity is,

$$sim_{sem}(v_i, v_j) = \frac{v_i \cdot v_j}{\|v_i\|\|v_j\|}$$

(5)

where $\|v_i\|$ and $\|v_j\|$ are the L^2 norm of v_i and v_j.

To measure the sequence similarity, the *Levenshtein Distance* is applied (Shi *et al.* 2014a). Let $|s_i|$ and $|s_j|$ denote the sequence length, the similarity between them is calculated by using,

$$sim_{seq}(s_i, s_j) = \begin{cases} 1 - \frac{dis_{Lev}(s_i, s_j)}{min(|s_i|, |s_j|)}, & \text{if } dis_{Lev}(s_i, s_j) \leq \\ & \varphi \cdot min(|s_i|, |s_j|) \\ 0, & \text{otherwise} \end{cases}$$

(6)

where $min(|s_i|, |s_j|)$ is the minimum length of s_i and s_j; φ is a threshold, $\varphi \in (0, 1)$.

By considering both semantic similarity and sequence similarity, the normalised similarity of projects p_i and p_j is measured by,

$$sim(p_i, p_j) = \\ \alpha \cdot sim_{sem}(v_i, v_j) + \beta \cdot sim_{seq}(s_i, s_j)$$

(7)

where α, β are the adjustable weights, $\alpha, \beta \in [0, 1], \alpha + \beta = 1$. These two variables are used to adjust the similarity measure process. For example, if $\alpha = 0$, the similarity between two projects is only determined by their content; if $\beta = 0$, the similarity between two projects is only determined by their processes. An empirical setting of them is $\alpha = \beta = 0.5$, which equally considers the effects from both content and processes.

Normality Measure

In the evaluation phase, *process normality* is used to evaluate whether the project process is on the right track during its operation (Shi *et al.* 2014a). In this paper, the normality measure of modelled processes is implemented on two levels, i.e., micro-level and macro-level.

Micro-level measure: for a given process-related feature, there could be different prior and posterior adjacent features in different project processes. The distribution of process-related features is mainly determined by two factors, feature property and project type. For the given feature, its own property determines its occurrence probability; the type of its related project determines its adjacent features. For example, for a process-related feature, named *'Issuing Repair Instruction'*, of In-Service projects, the prior features and posterior features, together with their occurrence probabilities are shown in Table 1 and Table 2 respectively.

Review of the data in these tables shows that certain feature combinations have higher occurrence probabilities, such as *'Request Repair Instruction'* and *'Issuing Repair Instruction'* (58.56%), *'Receiving Damage Report'* and *'Issuing Repair Instruction'* (14.38%), etc. However, some feature combinations have lower or zero occurrence probabilities, such as *'Issue Technical Disposition'* and *'Issuing Repair Instruction'* (2.74%), *'Issue Approval Sheet'* and *'Issuing Repair Instruction'* ($\simeq$0%). In general, if a project process mainly contains feature combinations with high occurrence probabilities, the process could be considered as normal; if the project process excessively contains feature combinations with low or zero occurrence probabilities, the process could be considered as less normal.

Table 1. List of prior features with probabilities

Prior Feature	Feature	Prob.
Request Repair Instruction		58.56%
Receive Damage Report	Issue	14.38%
Request Approval Sheet	Repair Instruction	9.70%
Request Damage Report		5.82%
Issue Technical Disposition		2.74%
...		...

In this micro-level measure, for a given dataset, the possible adjacent features of each single feature need to be identified, and the occurrence probabilities of the adjacent features need to be calculated accordingly. For a single feature $f_{P,x}$ and a dataset D, $Pr(f_{P,x})$ represents the feature

Table 2. List of posteriori features with probabilities

Feature	Posteriori Feature	Prob.
Issue Repair Instruction	Send Repair Instruction	46.15%
	Request Damage Report	7.69%
	Issue Approval Sheet	13.19%
	Receive Damage Report	12.09%
	Issue Technical Disposition	3.37%
	...	...

occurrence probability in the dataset, and $F_{x,D}$ represents a set of adjacent features of $f_{P,x}$ which are identified from the dataset. For a project containing the feature combination $(f_{P,x}, f_{P,x+1})$, its micro-level normality is measured by,

$$nor_{mic}(f_{P,x}, f_{P,x+1}) = \begin{cases} Pr(f_{P,x+1}), & \text{if } f_{P,x+1} \in F_{x,D} \\ -1, & \text{otherwise} \end{cases} \tag{8}$$

According to this equation, if a modelled process containing $(f_{P,x}, f_{P,x+1})$, but the feature $f_{P,x+1}$ never appears behind $f_{P,x}$ in the given dataset, then the normality score of this process will be decreased during the process evaluation.

Macro-level measure: the macro-level measure considers the entire project process, instead of the feature combinations. Fundamentally, giving a modelled process, this approach aims to calculate the number of other processes from the dataset that are similar to the given process.

Let s_i denote the entire sequence of a modelled process, $S_{i,D}$ is the set of other similar sequences contained in the dataset. The macro-level normality of the sequence is measured by,

$$nor_{mac}(s_i) = \begin{cases} |S_{i,D}| \cdot \sum_{j=1}^{|S_{i,D}|} sim_{seq}(s_i, s_j) \\ 0, & \text{if } \frac{|S_{i,D}|}{|D|} \leq \varphi \end{cases} \tag{9}$$

where $|S_{i,D}|$ is the size of $S_{i,D}$, i.e., the number of similar sequences of s_i in dataset D; $|D|$ is the total number of sequences contained in the dataset. φ is a threshold, $\varphi \in (0,1]$, which is used to adjust the process of macro-level measure. For example, when $\varphi = 0.05$, the sequence s_i will be considered as normal, if and only if its similar sequences take more than 5% proportion in the dataset.

Evaluation and Experiment

This section demonstrates the functionalities of APAS and evaluates the analytical modules of the system through the use of a dataset from an aerospace organisation. The data set contained 396 In-Service projects that were implemented between 2013 and 2014. The data of each project included workflow, technical and communication related information. All of the information items had identifiable timestamps. The knowledge about general operation process and feature weightings was captured from the subject matter experts in the departments. It is then used as the knowledge bases to support related analytical modules within APAS. The evaluation comprised of three main parts, i) data processing and feature modelling, ii) process modelling and monitoring, and iii) process evaluation.

Data Processing and Feature Modelling

The data of In-Service projects included various file types e.g., database related (.sql), image related (.jpg, .tiff, scanned PDF files) and text related (.pdf, .doc, .txt). These types of data are common across a lot of organisations and industries. Some of the image files also contained textual information e.g., scanned PDFs, drawings or images with annotations. To utilise the information contained in such files, optical character recognition was used to pre-process some of the data. After this, all the textual data was processed using natural language processing. The processing method involved tokenisation, stop words removing and stemming, which filtered out the terms with less importance, and re-organised the data into a more structured format.

Semantic analysis with lexical ontologies is used to extract semantic features from the pre-processed data. The main reason of using lexical ontologies is to address the polysemy and synonymy problems. In general, polysemy means that one term could have different concepts, and synonymy means that one concept could be described by different terms. For example, 'fuselage' and 'airframe' have the similar semantic meaning but in different forms. For the information items containing 'fuselage' and 'airframe' respectively, they could be considered to have the identical concept by the project actors straightaway. However, they could be considered to have different concepts by the system if only term-based analysis is applied. Eliminating disambiguated word senses is recognised as the essential way for improving the accuracy of content analysis (Agirre & Edmonds 2007). For information management systems in manufacturing, correctly distinguishing the semantic meaning of project data is critical, as it can directly affect the way of organising unstructured information, discovering knowledge and sharing information between people/departments (Xie *et al.* 2011).

After the initial processing, the project data is converted into a set of semantic features. Table 3 shows the semantic features being extracted from 'ODR' (Operator's Damage Report). As shown in the table, the semantic features are the essential terms contained in the report content, and the importance of feature is indicated by the feature occurrence. The semantic features with their occurrences describe the project details, service requirements and technical requirements. For example, the features from the table could describe the affected locations, i.e.,'wing', 'top skin', 'rear rib', or service types, i.e., 'cracking, 'corrosion'.

Besides the semantic features, the process-related features also need to be identified and extracted. Table 4 shows a list of process-related features that are extracted from the content and metadata of ODR. In the feature modelling module, named entity recognition and information extraction are applied to perform this task. Named entity recognition could automatically identify named entities from the data, and the identified entities include people, organisation, date, location, etc. Information extraction is applied to detect the relation/dependency between the identified entities. With the assistance of pre-defined rule sets and knowledge bases, information extraction also identifies the information with more specific properties, including aircraft types, part names, serial numbers, document reference numbers and contact information.

Table 3. Semantic features of the ODR

Term	Occurrence	Term	Occurrence
top	102	damage	37
skin	87	threshold	35
rear	81	location	34
rib	79	j1	32
hole	75	fh	31
crack	72	face	31
fastener	62	rh	30
issue	59	structure	27
wing	56	fit	25
fatigue	55	bay	25
flange	49	justification	25
stress	47	material	25
inspection	47	load	24
corrosion	43	...	...

From a general information management point of view, converting heterogeneous data into appropriate features could facilitate content analysis and feature modelling. It enables the system to integrate with different analytical technologies, including content retrieval, content-based clustering/classification and sequential pattern mining. Moreover, the selection and utilisation of knowledge bases will be performed more efficiently and effectively further improving the automation and performance of the analytical modules.

Process Modelling and Monitoring

Once the data processing and feature identification have been completed, each project process is then represented by a set of combined features, i.e., semantic features and process-related features. The semantic features need to be converted into feature vectors. Table 5 shows the feature vectors that are generated from part of the dataset. In this table, each row is the feature vector of one project process.

To generate the sequence, process-related features are applied. Certain features may have higher importance than the others, if they have higher correlations with the key activities in the project process. The information contained in these features is sufficient to represent the project process, thus the process modelling module only needs to take into account of certain features, instead of the entire feature set. It also means that some features with less importance should be removed deliberately. This could improve the robustness and rationality of the process modelling module, as well as eliminate the possible interferences caused by irrelevant information. Based on the knowledge captured from the In-Service departments, 15 types of process-related features are considered, and some of them are shown in Table 6.

For any given dataset, the process modelling module analyses its contained project information recursively, and generates separate feature set for each project, whilst the timestamp of each feature is identified. The considered features (in Table 6) with timestamps are then used to create the process sequences. Figure 5 shows a visualisation of some modelled processes. In this visualisation, each row is a modelled project process. Each *Tx* indicates a single feature, and the features contained in a sequence are arranged in chronological order. This visualisation helps project actors review multiple modelled process simultaneously, which

enables them to efficiently gain the general understanding of the process structure.

Moreover, Figure 6 shows another visualisation of the modelled processes based on an interactive interface. Comparing to the previous one, this visualisation shows the explicit relationship between each feature and its timestamp. It enables the project actors to have a drill-down view of each modelled process on document or content level, which aims to help them to understand the process inner structure and the dependency of the features. For example, EMAIL indicates the initialisation of a task; LINEITEM indicates the type of action being required by the task implementation; and ATTACHMENT indicates the actual documents or data being generated during the task implementation. All the features are then organised into groups based on their timestamps, which can be zoomed in or zoomed out by using the navigation tabs on the interface. Both listed visualisations (Figure 5 and 6) are included in the *process visualisation* module.

Process Evaluation

This evaluation aims to investigate the process evaluation module through two scenarios, i.e., unsupervised process evaluation and supervised process evaluation.

Unsupervised Process Evaluation

The process evaluation module contains automatic mechanisms of sequence analysis and process normality identification. In the unsupervised scenario, advanced domain knowledge, labelled or pre-classified data are not necessary for performing sequence analysis and normality identification. The process normality of projects is measured based on the sequence structure and the dataset structure. For the given dataset, the sequence similarity of each project is calculated by using the proposed approach, and then a similarity matrix is generated accordingly (as shown in Table 7). The data shown in the matrix is from Figure 5*. To simplify it, sequence similarity values less than 0.5 are filtered out.

In the matrix, projects P2 and P3 are considered the most similar, with the highest similarity value 0.7185. According to their sequence structures shown in Figure 5, some common patterns can be identified in both of them. For example, in the early project stage, pattern {T1, T9} occurred; in the middle project stage, pattern {T1, T9, T13, T15, T1} occurred; and in the late project stage, patterns {T9, T11} and {T15, T13, T15, T9} occurred. By examining their content-related information, both projects are related to an identical aircraft model, same damage location and service type. Moreover, projects P5, P6 and P8 are all considered to be unique, as their similarity values are the lowest ones in the matrix. According to their sequence structures, few common patterns are contained in their processes when comparing to the others. By examining their content-related information, project P5 is the only one having 'lightening' related damage, therefore its process should be different from others in natural. Project P6 is a new type of damage that has never occurred in the past, a special service procedure therefore needs to be designed for it, leading to its process being different from the others. Project P8 has a different

---

*Due to space limitation, only the first 10 entries are displayed in the matrix.

Table 4. Identified process-related features of the ODR

'feature_name': 'ODR', 'feature_id': '1', 'feature_property': 'outgoing', 'file_type': ['text', 'image',], 'aircraft_type': 'A320-XX',	'project_actors': [P1, P2,], 'departments': [D1, D2,], 'location': [L1, L2,],	'activated_date': ['2013-04-15', '2013-04-16', '2013-04-22',], 'document_refs': ['7057597x/0x4', '7057597x/0x9', '7057597x/0x4', '7057597x/0x2',], ...

Table 5. Sample of semantic feature vectors

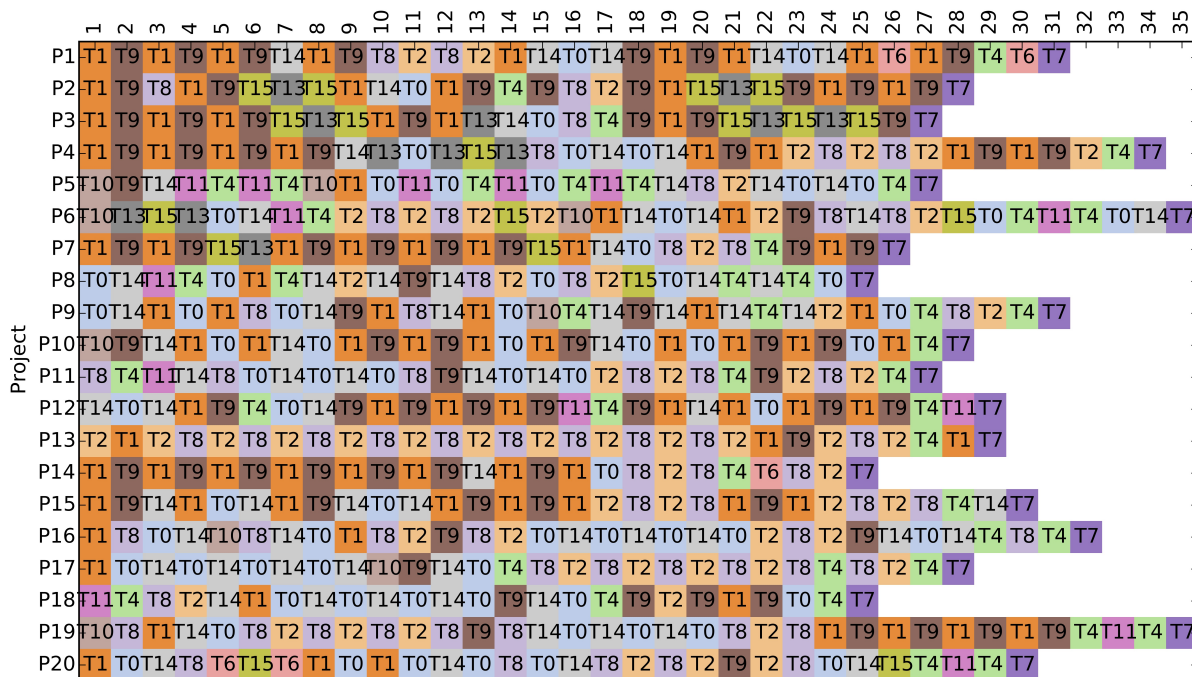|	skin	lh	multisite	lightning	rear	upper	rh	panel	surface	rib	...
P1	0.087	0	0.113	0	0	0	0.087	0.087	0	0	...
P2	0.066	0	0	0	0.087	0	0	0	0	0.066	...
P3	0.080	0	0.105	0	0	0	0	0.080	0	0.080	...
P4	0.066	0.087	0	0	0	0	0	0.066	0.066	0.066	...
P5	0	0	0	0.210	0	0	0.080	0	0	0	...
P6	0.075	0	0	0	0.098	0.098	0	0.075	0.075	0	...
P7	0.075	0	0	0	0	0.098	0	0.075	0.075	0.075	...
P8	0.070	0	0	0	0	0	0.070	0	0.070	0	...
P9	0	0.123	0	0	0	0	0	0	0	0	...
P10	0.087	0	0	0	0	0.113	0	0	0	0	...

Figure 5. Sample of process sequences

damage location, i.e., its damage is on the 'lower surface' whereas the others are on the 'top surface', hence a different process is required.

Comparing process normality for large number of modelled processes by the use of such detailed information is not always realistic, due to limited time and restricted access to the information in practice. The macro-level normality is considered to be a fast approach to comparing multiple processes more efficiently. It takes into account of both the similarity values and the quantity of similar projects. The macro-level normality of a modelled process is in proportion to the cumulative similarity of its related projects, in conjunction with the number of its related projects. For example, project P7 has 5 related projects, and its cumulative similarity is equal to $(0.546 + 0.662 + 0.585 + 0.508 + 0.585) = 2.886$, thus its macro-level normality is $2.886 \times 5 = 14.430$. In general, for two projects having the identical cumulative similarity, the one with more similar
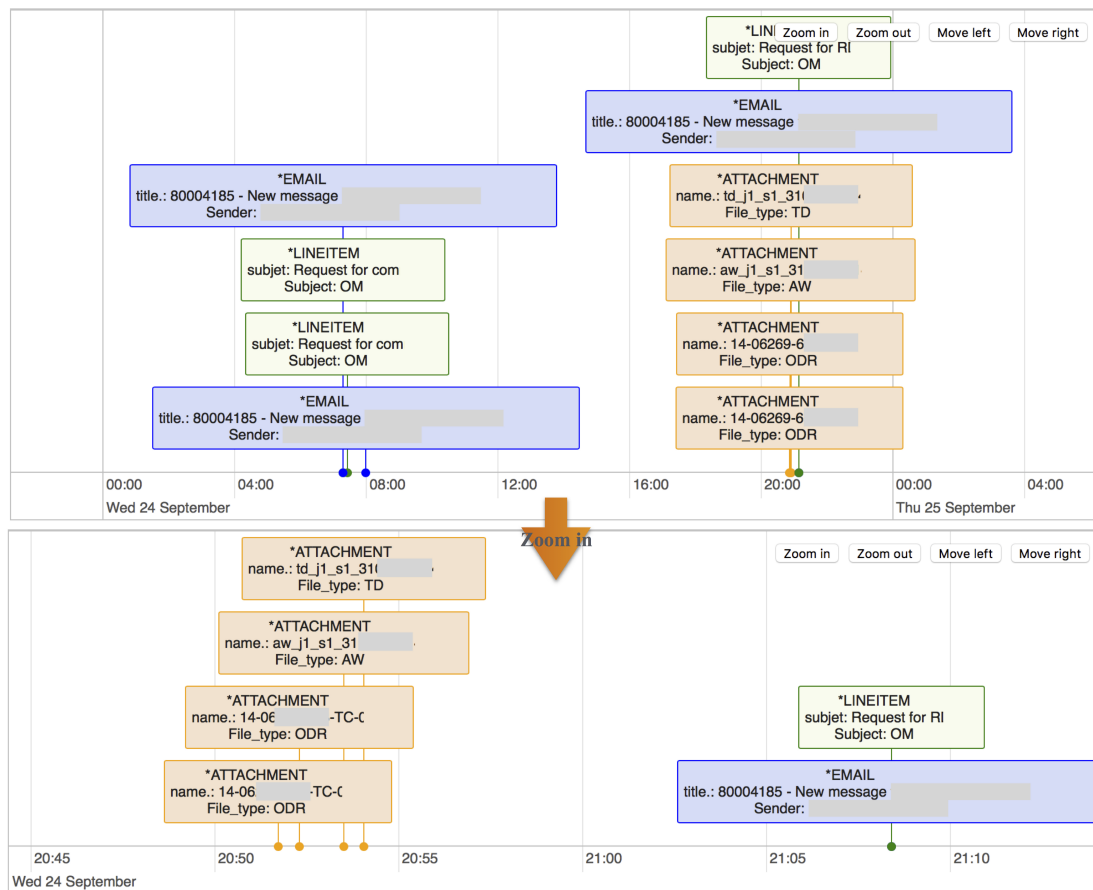
Figure 6. An interactive interface for visualising the modelled processes

Table 6. List of process-related features

Feature
OM Outgoing Message
AW Answer
S&F Stress & Fatigue
ODR Operator's Damage Report
IM Incoming Message
RDAS Repair/Design Approval Sheet
TD Technical Disposition
DRG Drawing
RI Repair Instruction
...

projects will be considered to have higher normality; for two projects having the same number of similar projects, the one with higher value of cumulative similarity will be considered to have higher normality.

Supervised Process Evaluation

In the supervised scenario, the process evaluation module assesses the characteristics of new projects by using some labelled data. In practice, the departments may already have the labelled/annotated data of completed projects. By the use of past experience and modelled knowledge, more labelled data can be created automatically using data mining technologies. In this evaluation, six labels are assigned to the projects in the dataset, according to the specific rules summarised by knowledge experts of the departments. Projects with an identical label means they are similar

to each other in terms of their contained process-related features. Next, 10-fold cross-validation is applied to divide the dataset into a training set (containing 90% data) and a test set (containing 10% data). Afterwards, some *classification* approaches, i.e., Naive Bayes, Artificial Neural Network (ANN), Support Vector Machine (SVM) and Random Forest (RF), are applied to classify the test set based on the training set. The objectives of this evaluation include, i) testing if the labelled projects can be used to evaluate new projects (the label of test data are invisible to these classifiers, thus the test data can be treated as 'new projects' in the evaluation), ii) testing if the data representation proposed in this paper has the capability to work with standard classification approaches. The results of this evaluation are shown in Table 8.

Overview

According to the results from the table, the classification approaches, especially ANN and RF, have good performance on classifying unlabelled project processes. This indicates that new project processes can be automatically classified based on the labelled processes with appropriate accuracy. Meanwhile, the results also demonstrate that the proposed data representation can work with standard classification approaches with no issues. Moreover, the evaluation in the unsupervised scenario confirms that the process with low normality scores can be automatically identified by applying the proposed approach. All these results show that the

Table 7. Sequence similarity and normality

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
P1		0	0	0.619	0	0	0.546	0	0.523	0.557
P2	0		0.719	0	0	0	0.662	0	0	0.593
P3	0	0.719		0	0	0	0.585	0	0	0
P4	0.620	0	0		0	0	0.508	0	0	0
P5	0	0	0	0		0	0	0	0	0
P6	0	0	0	0	0		0	0	0	0
P7	0.546	0.662	0.585	0.508	0	0		0	0	0.585
P8	0	0	0	0	0	0	0		0	0
P9	0.523	0	0	0	0	0	0	0		0.557
P10	0.557	0.593	0	0	0	0	0.585	0	0.557	
M-Nor.	8.980	5.919	2.606	2.254	0.000	0.000	11.425	0.000	2.160	9.168

Table 8. Results of process classification

	Naive Bayes	ANN	SVM	RF
Precision	0.792	0.933	0.859	0.915
Recall	0.794	0.932	0.841	0.916
F-measure	0.790	0.932	0.830	0.915

process evaluation module in the system has the capability to evaluate the project processes.

Conclusions

The proposed system APAS in this paper aims to support the automation of process management for collaborative engineering projects. The main functionalities of the system include process modelling, process monitoring and process evaluation. To automate the system functionalities, related analytical approaches have been proposed, including feature modelling, process sequencing, process similarity measure and process normality identification. Using the proposed approaches, the system converts project data into processes that can be modelled, in the form of feature vectors and sequences. It is then possible to perform automatic analysis on them. Thus the approach and the associated system enables project actors to monitor and evaluate multiple project processes simultaneously, whilst giving them unique understanding on process evolutions, project characteristic changes and environment dynamics.

The evaluation shows that the system with proposed approaches have the capability to work with real engineering data. It also shows that the proposed data representation for engineering projects can be integrated with various data mining technologies, implying that the functionality of this process management system can be customised and extended.

Further work includes the development of a knowledge capture module. This module is required to be integrated with the knowledge bases. The main functionalities include the detection of common knowledge of historical projects, and the reduction of human intervention for knowledge base creation. In addition, the design of a formal knowledge structure for aerospace In-Service is under consideration, and the evaluation of such knowledge structure is also required.

Acknowledgements

References

Abdalla, Hassan S. 1999. Concurrent engineering for global manufacturing. *International Journal of Production Economics*, **60**, 251–260.

Agirre, Eneko, & Edmonds, Philip Glenny. 2007. *Word sense disambiguation: Algorithms and applications.* Vol. 33. Springer Science & Business Media.

Agrawal, Divyakant, El Abbadi, Amr, Antony, Shyam, & Das, Sudipto. 2010. Data management challenges in cloud computing infrastructures. *Pages 1–10 of: Databases in Networked Information Systems.* Springer.

Airbus. 2002. Less maintenance, less costs. *FAST magazines*.

Airbus. 2006. Maintenance cost and reliability control. *FAST magazines*.

Airbus. 2014. Flying on demand. *Airbus' Global Market Forecast for 2014-2033*.

Al-Najjar, Basim, & Alsyouf, Imad. 2004. Enhancing a company's profitability and competitiveness using integrated vibration-based maintenance: A case study. *European Journal of Operational Research*, **157**(3), 643 – 657.

Benner, Mary J., & Tushman, Michael. 2002. Process Management and Technological Innovation: A Longitudinal Study of the Photography and Paint Industries. *Administrative Science Quarterly*, **47**(4), 676–707.

Carey, E., Culley, S., & Weber, F. 2013. Establishing key elements for handling in-service information and knowledge. *Pages 11–20 of: Proceedings of the International Conference on Engineering Design, ICED. Vol. 6 DS75-06.* The Design Society.

Caton, Simon, & Rana, Omer. 2012. Towards autonomic management for cloud services based upon volunteered resources. *Concurrency and Computation: Practice and Experience*, **24**(9), 992–1014.

Choudhary, Alok Kumar, Harding, Jenny A, & Tiwari, Manoj Kumar. 2009. Data mining in manufacturing: a review based on the kind of knowledge. *Journal of Intelligent Manufacturing*, **20**(5), 501–521.

de Mast, Jeroen, Kemper, Benjamin, Does, Ronald J. M. M., Mandjes, Michel, & van der Bijl, Yohan. 2011. Process improvement in healthcare: overall resource efficiency. *Quality and Reliability Engineering International*, **27**(8), 1095–1106.

de Medeiros, Ana Karla Alves, Van der Aalst, Wil, & Pedrinaci, Carlos. 2008. Semantic process mining tools: core building blocks.

Deelman, Ewa, & Chervenak, Ann. 2008. Data management challenges of data-intensive scientific workflows. *Pages 687–692 of: Cluster Computing and the Grid, 2008. CCGRID'08. 8th IEEE International Symposium on*. IEEE.

Dobson, Simon, Sterritt, Roy, Nixon, Paddy, & Hinchey, Mike. 2010. Fulfilling the vision of autonomic computing. *Computer*, 35–41.

Dychtwald, Ken, Erickson, Tamara J, & Morison, Robert. 2013. *Workforce crisis: How to beat the coming shortage of skills and talent*. Harvard Business Press.

Eppler, Martin J., & Mengis, Jeanne. 2004. The Concept of Information Overload: A Review of Literature from Organization Science, Accounting, Marketing, MIS, and Related Disciplines. *The Information Society*, **20**(5), 325–344.

Fallon, Liam, & O'Sullivan, Declan. 2013. Aesop: A semantic system for autonomic management of end-user service quality. *Pages 716–719 of: Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on*. IEEE.

Gröger, Christoph, Niedermann, Florian, & Mitschang, Bernhard. 2012. Data mining-driven manufacturing process optimization. *Pages 4–6 of: Proceedings of the World Congress on Engineering*, vol. 3.

Haddar, Nahla Zaaboub, Makni, Lobna, & Abdallah, Hanene Ben. 2014. Literature review of reuse in business process modeling. *Software & Systems Modeling*, **13**(3), 975–989.

Harding, JA, Shahbaz, M, Kusiak, A, *et al.* . 2006. Data mining in manufacturing: a review. *Journal of Manufacturing Science and Engineering*, **128**(4), 969–976.

Hislop, Donald. 2013. *Knowledge management in organizations: A critical introduction*. Oxford University Press.

Hörisch, Jacob, Johnson, Matthew P., & Schaltegger, Stefan. 2014. Implementation of Sustainability Management and Company Size: A Knowledge-Based View. *Business Strategy and the Environment*.

Huebscher, Markus C, & McCann, Julie A. 2008. A survey of autonomic computing—degrees, models, and applications. *ACM Computing Surveys (CSUR)*, **40**(3), 7.

IBM. 2006. An architectural blueprint for autonomic computing. *IBM White Paper*.

Kephart, Jeffrey O., & Chess, David M. 2003. The Vision of Autonomic Computing. *Computer*, **36**(1), 41–50.

Kim, Hyunjoo, El-Khamra, Yaakoub, Rodero, Ivan, Jha, Shantenu, & Parashar, Manish. 2011. Autonomic management of application workflows on hybrid computing infrastructure. *Scientific Programming*, **19**(2-3), 75–89.

Ko, Ryan K. L. 2009. A Computer Scientist's Introductory Guide to Business Process Management (BPM). *Crossroads*, **15**(4), 4:11–4:18.

Köksal, Gülser, Batmaz, İnci, & Testik, Murat Caner. 2011. A review of data mining applications for quality improvement in manufacturing industry. *Expert systems with Applications*, **38**(10), 13448–13467.

Lerner, Barbara Staudt, Christov, Stefan, Osterweil, Leon J., Bendraou, Reda, Kannengiesser, Udo, & Wise, Alexander. 2010. Exception Handling Patterns for Process Modeling. *IEEE Transactions on Software Engineering*, **36**(2), 162–183.

Lewin, Arie Y, Massini, Silvia, & Peeters, Carine. 2009. Why are companies offshoring innovation? The emerging global race for talent. *Journal of International Business Studies*, **40**(6), 901–925.

Lindner, Frank, & Wald, Andreas. 2011. Success factors of knowledge management in temporary organizations. *International Journal of Project Management*, **29**(7), 877 – 888. Complexities in Managing Mega Construction Projects.

Michael Gnatz, Martin Deubler, Michael Meisinger, & Rausch, Andreas. 2004. Towards an integration of process modeling and project planning. *IET Conference Proceedings*, 22–31(9).

Ming, X.G., Yan, J.Q., Wang, X.H., Li, S.N., Lu, W.F., Peng, Q.J., & Ma, Y.S. 2008. Collaborative process planning and manufacturing in product lifecycle management. *Computers in Industry*, **59**(2–3), 154 – 166.

Müller, Dominic, Herbst, Joachim, Hammori, Markus, & Reichert, Manfred. 2006. IT Support for Release Management Processes in the Automotive Industry. *Pages 368–377 of: Proceedings of the 4th International Conference on Business Process Management*. BPM'06.

Nieusma, Dean, & Riley, Donna. 2010. Designs on development: Engineering, globalization, and social justice. *Engineering Studies*, **2**(1), 29–59.

Patelli, Alina, Calinescu, Radu, & Wang, Hai. 2014. Semantic Reasoning for Autonomic IT Systems. *Pages 13–18 of: Proceedings of the 19th International Doctoral Symposium on Components and Architecture*. WCOP '14.

Pino, Francisco J., García, Félix, & Piattini, Mario. 2008. Software Process Improvement in Small and Medium Software Enterprises: A Systematic Review. *Software Quality Control*, **16**(2), 237–261.

Rao, Lila, Mansingh, Gunjan, & Osei-Bryson, Kweku-Muata. 2012. Building ontology based knowledge maps to assist business process re-engineering. *Decision Support Systems*, **52**(3), 577 – 589.

Rosemann, Michael, & vom Brocke, Jan. 2015. The six core elements of business process management. *Pages 105–122 of: Handbook on Business Process Management 1*. Springer.

Schäfermeyer, Dipl-Kfm Markus, Rosenkranz, Christoph, & Holten, Roland. 2012. The impact of business process complexity on business process standardization. *Business & Information Systems Engineering*, **4**(5), 261–270.

Scheer, August-Wilhelm, & Nüttgens, Markus. 2000. *ARIS architecture and reference models for business process management*. Springer.

Shen, Weiming, Wang, Lihui, & Hao, Qi. 2006. Agent-based distributed manufacturing process planning and scheduling: a state-of-the-art survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, **36**(4), 563–577.

Shi, Lei, Gopsill, James, Newnes, Linda, & Culley, Steve. 2014a. A Sequence-Based Approach to Analysing and Representing Engineering Project Normality. *Pages 967–973 of: Tools with Artificial Intelligence (ICTAI), 2014 IEEE 26th International Conference on*.

Shi, Lei, Gopsill, JA, Snider, CM, Jones, SL, Newnes, Linda, & Culley, SJ. 2014b. Towards Identifying Pattern in Engineering Documents to Aid Project Planning. *In: DS 77: Proceedings of the DESIGN 2014 13th International Design Conference*.

Sterritt, Roy, Parashar, Manish, Tianfield, Huaglory, & Unland, Rainer. 2005. A concise introduction to autonomic computing. *Advanced Engineering Informatics*, **19**(3), 181–187.

Van Der Aalst, Wil MP. 2013. Business process management: A comprehensive survey. *ISRN Software Engineering*.

Vianello, Giovanna, Xie, Yifan, Ahmed-Kristensen, Saeema, Culley, Stephen J, *et al.* . 2010. Handling of in-service support: comparison of two case studies from complex industries. *In: DS 60: Proceedings of DESIGN 2010, the 11th International Design Conference, Dubrovnik, Croatia.*

Wang, Kesheng. 2007. Applying data mining to manufacturing: the nature and implications. *Journal of Intelligent Manufacturing*, **18**(4), 487–495.

Weber, Frithjof, Dauphin, Eva, Fuschini, Renata, Haarmann, Julia, Katzung, Alexander, & Wunram, Michael. 2007. Expertise transfer: A case study about knowledge retention at Airbus. *Proceedings of the 13th International Conference on Concurrent Enterprising (ICE 2007)*, 329–338.

Weske, Mathias. 2012. *Business process management: concepts, languages, architectures*. Springer Science & Business Media.

Xie, Yifan, Culley, Steve, & Weber, Frithjof. 2011. Applying context to organize unstructured information in aerospace industry. *In: DS 68-6: Proceedings of the 18th International Conference on Engineering Design (ICED 11), Impacting Society through Engineering Design, Vol. 6: Design Information and Knowledge, Lyngby/Copenhagen, Denmark.*