AI Magazine Volume 29 Number 1 (2008) (© AAAI)

An AI Framework for the Automatic Assessment of e-Government Forms

Andy Hon Wai Chun

■ *This article describes the architecture and AI technology behind an XML-based AI framework designed to streamline e-government form processing. The framework performs several crucial assessment and decision support functions, including workflow case assignment, automatic assessment, follow-up action generation, precedent case retrieval, and learning of current practices. To implement these services, several AI techniques were used, including rule-based processing, schema-based reasoning, AI clustering, case-based reasoning, data mining, and machine learning. The primary objective of using AI for e-government form processing is of course to provide faster and higher quality service as well as ensure that all forms are processed fairly and accurately. With AI, all relevant laws and regulations as well as current practices are guaranteed to be considered and followed. An AI framework has been used to implement an AI module for one of the busiest immigration agencies in the world.*

Government agencies around the world process thousands to millions of forms yearly; some may even process a million forms a day (VertMarkets IT Group 2002). To improve efficiencies and reduce cost, many government agencies use some type of document imaging solution for unstructured information or form-processing solutions to capture structured data (Captiva Software Corporation 2002, VertMarkets IT Group 2003). Some government agencies also offer e-forms as an alternative to paper forms to their citizens for even more efficient and accurate data collection.

Once forms are digitized and indexed and the data extracted, it is just a natural progression to consider using artificial intelligence (AI) to further enhance efficiencies with intelligent decision support. This article describes use of an XML-based AI framework to create an AI module for an immigration agency to support its extensive form-processing needs.

Immigration agencies play vital roles in maintaining the security and prosperity of a place. They control the entry and departure of people at its borders and safeguard it against threats. They may also be responsible for enforcing immigration control within the boundaries of the place. Besides immigration control, the immigration agency for which this work was performed is also responsible for providing a wide variety of document-related services to its citizens and visitors. These services include issuing various types of travel documents, identity cards, nationality documents, visas or permits, right of abode, and birth, death, and marriage registrations. In fact, the agency routinely handles more than a hundred different types of document requests. In 2004, the agency processed close to 4 million application forms at its headquarters, which has a tight workforce of roughly a couple of thousand.

To overcome rapidly increasing workloads, the agency looks toward IT to improve efficiency and productivity (Hong Kong Trade Development Council 2006, Hong Kong Government 2005). The AI project described in this article is part of a new IT strategy to streamline the entire immigration form-processing workflow with advanced document management and forms processing software (Questex Media 2006, Hong Kong Government 2004). The new system provides the agency with virtually a paperless environment where all documents are digitized and indexed with automatic data extraction from forms. The AI module further streamlines processes and workflows with decision support capabilities to help the agency cope with continued growth.

With the new system, the public will be able to obtain services by submitting either paper forms or e-forms (for some services). With AI decision support, visits to the agency will be minimized, and processing time will be significantly shortened. One-stop, on-the-spot service will be provided for some application types. Application status and progress can be checked through the web. Overall, the new IT strategy greatly improves the level of convenience to citizens and visitors.

Current Manual Approach

The workflow for each type of form may be slightly different. Here, I will describe a typical workflow for the manual approach (figure 1 is a simplified process flow diagram). This workflow is probably similar to those of other government agencies around the world.

The process starts with the applicant appearing in person to submit paper forms together with relevant documents and papers. The frontline staff at the counter does a preliminary check to see if all the necessary documents are attached, and the applicant leaves.

A case folder is created and eventually passed to an authorization officer who will do a preliminary assessment of the case and then assign a suitable case officer actually to process the case to completion. The case officer is assigned according to his or her experience and familiarity with handling that type of application. After a thorough and detailed review and analysis of the application form, the case officer may request additional supplementary documents from the applicant. Several rounds of visits may be needed depending on the content of the documents provided and the nature of the application. For example, it may be the case that the applicant might not qualify for the given application type but may qualify under a different scenario. If so, different sets of documentation may be required. The case officer may need to consider multiple approval scenarios at the same time.

When all the supporting documents have been submitted and verified, the case officer will make a final assessment, which will then be reviewed and endorsed by the authorization officer. Finally, the applicant will be notified of the result and return to collect the requested documents or permits if application was successful. The entire process may require several visits by the applicant to the immigration agency and many days, weeks, or even months to complete depending on complexity.

In order for a case officer to adequately process an application, he or she must possess thorough knowledge of all the applicable laws and regulations as well as immigration guidelines, which might change from time to time. In addition, the case officer must also be able to use his or her experience in processing other similar cases to draw on precedent cases for reference if discretionary decision making is needed. Historical case documents are available in microfiche, but searching for related cases will take time. The case officer may need to consult with other more senior or experienced case officers before a decision can be made. As you can see, assessing a complex case, such as applying for right of abode, can be very time consuming and knowledge intensive.

AI Project Objectives

Very challenging goals were defined for the AI system—to streamline the entire assessment workflow with automated decision support wherever possible. The key objectives for the AI module are (1) to automatically assess straightforward cases, (2) to provide decision support for nonstandard cases, and (3) to learn “current practices” from humans.

Cases are divided into straightforward cases and nonstandardcases. Straightforward cases are those for which a determination as to whether they satisfy applicable laws and regulations can be made immediately and require very little processing. Nonstandard cases are those that may require additional information or documentation or may involve discretionary decision making by the case officer. Discretionary decision making must follow current practices and guidelines. Since practices and guidelines change from time to time to reflect changing needs, the AI module will need to automatically adapt itself through learning.

New AI Approach

Based on these AI goals and objectives, several new AI processes were designed to streamline the forms processing workflow (see figure 2—processes A1 to A6). With the new AI system, application forms will either be submitted online or as hard copy and then scanned and processed by optical character recognition. Associated supporting documents will also be stored digitally in a secured document management system. For simple forms, online submission represents substantial savings in community cost since the applicant need not go to the agency in person (Hong Kong Government 2004).

After submission, the AI module assists with case assignment (process A1 in figure 2) by automatically categorizing the case into defined categories. At the same time, it performs an initial case assessment (A2) that is used by the assigned case officer to determine whether the case is a straightforward or a nonstandard case. Case assessment is done by evaluating the case against all applicable laws and

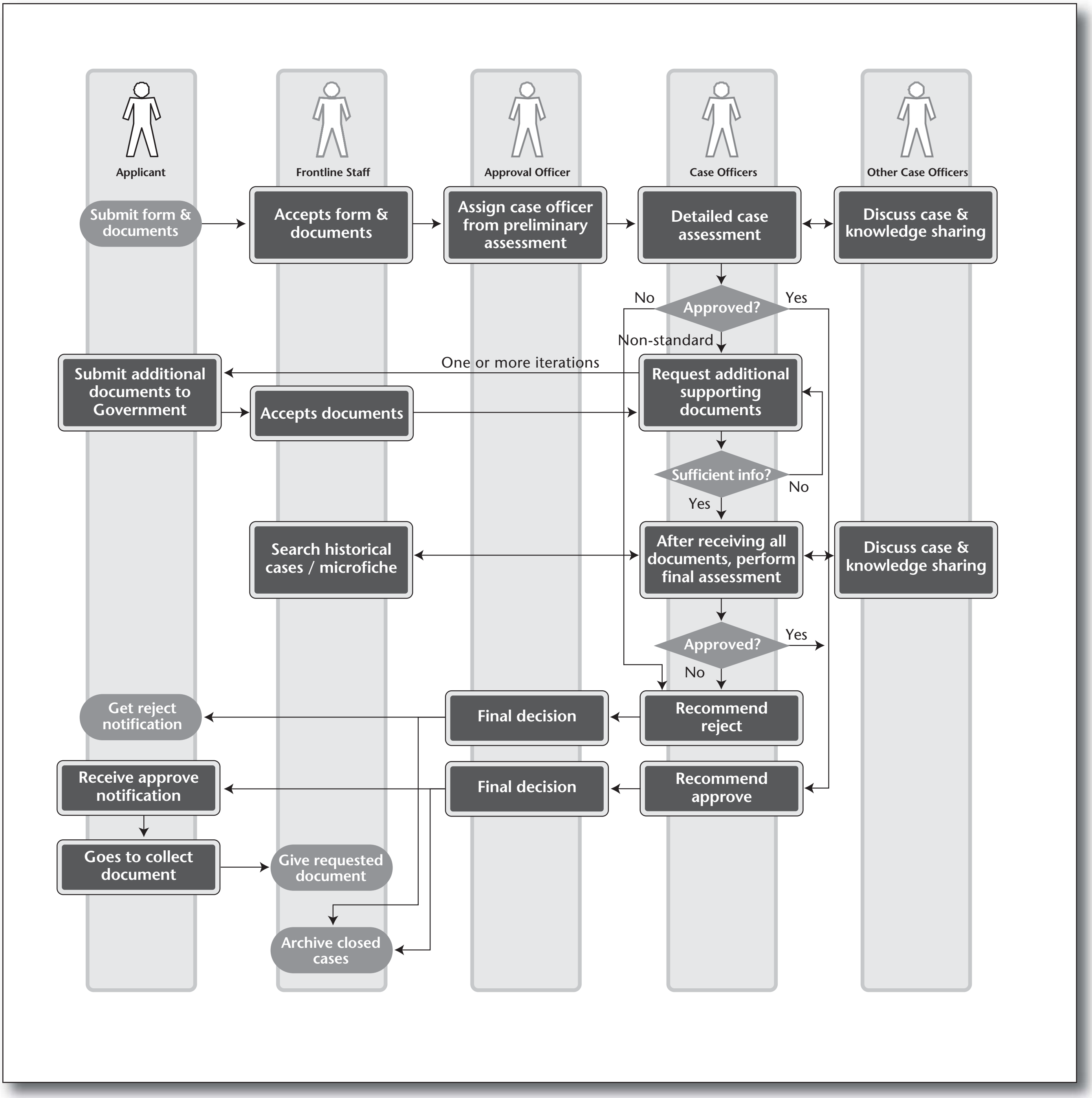

Figure 1. Process Flow Diagram for Manual Approach.

regulations as well as current practices and guidelines for each application type. For certain types of application, the assessment may be done in a "one-stop" fashion and the applicant can collect the permits or letters during the same visit. Since for many application types, a majority of the cases are straightforward cases, the agency estimates great efficiency savings with AI.

For nonstandard cases, the case officer will use the AI module to (A3) generate follow-up actions—these are suggested steps to take in order to get the application to a final state that can be assessed. For example, the AI module may recommend that the case officer request additional supplementary documents or clarifications from the applicant.

In the manual approach, follow-up actions may be an iterative process. The applicant may need to visit the agency more than one time before all the

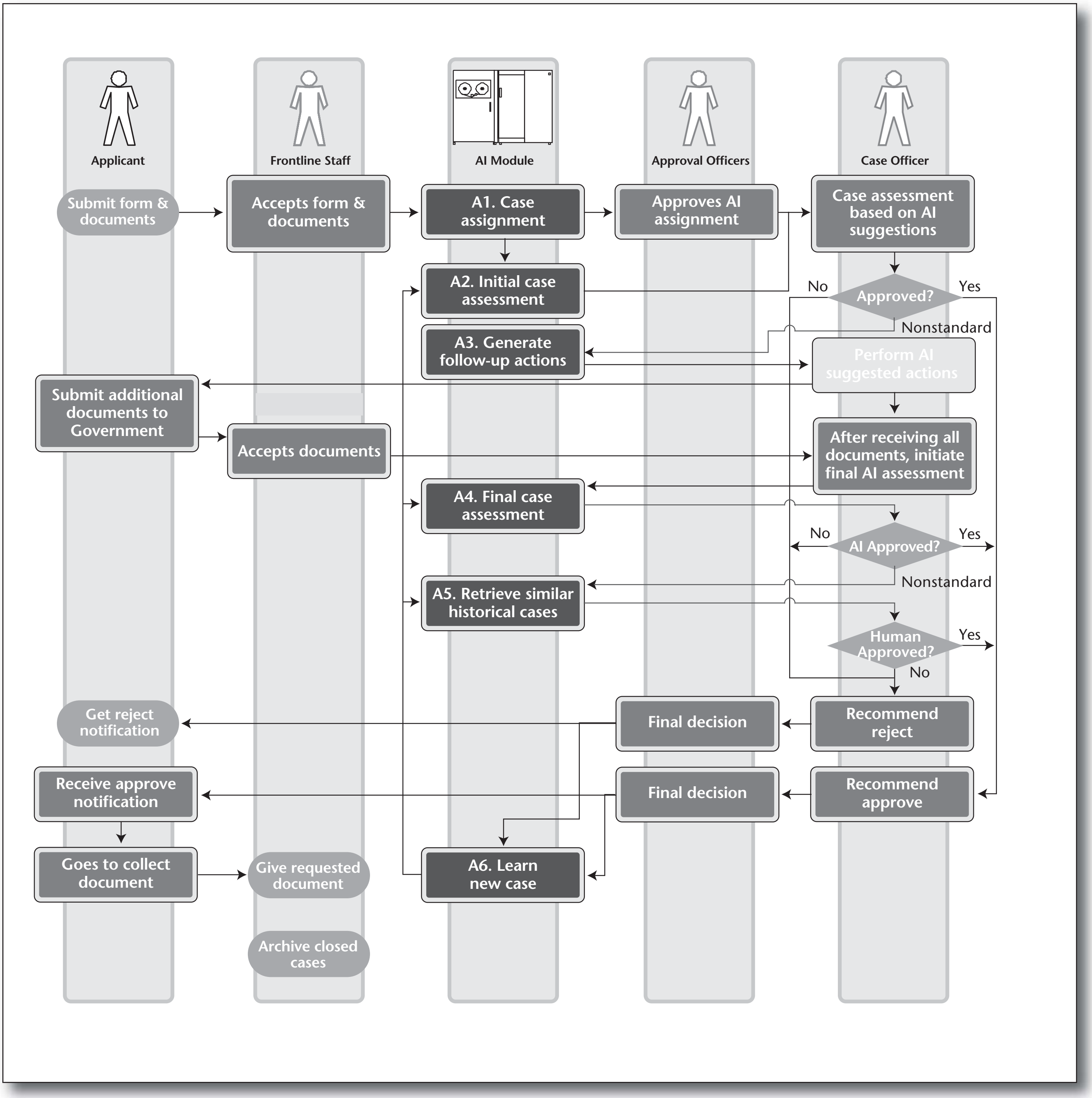

Figure 2. Process Flow Diagram for AI Approach.

necessary documents are collected and information clarified. With AI, all possible approval scenarios are considered at the same time before follow-up actions are generated, thus reducing the number of visits needed.

Once information is complete, the case officer will request the AI module to perform (A4) final case assessment. This is similar to (A2) initial case assessment except that now information is complete.

For complex borderline cases, some form of discretionary decision making may be required from the case officer. The AI module assists this process by (A5) retrieving a set of “similar” cases from historical records together with assessment results and justifications for the case officer to use as reference. This is done within seconds compared with hours or possibly days to search through microfiche to find reference cases.

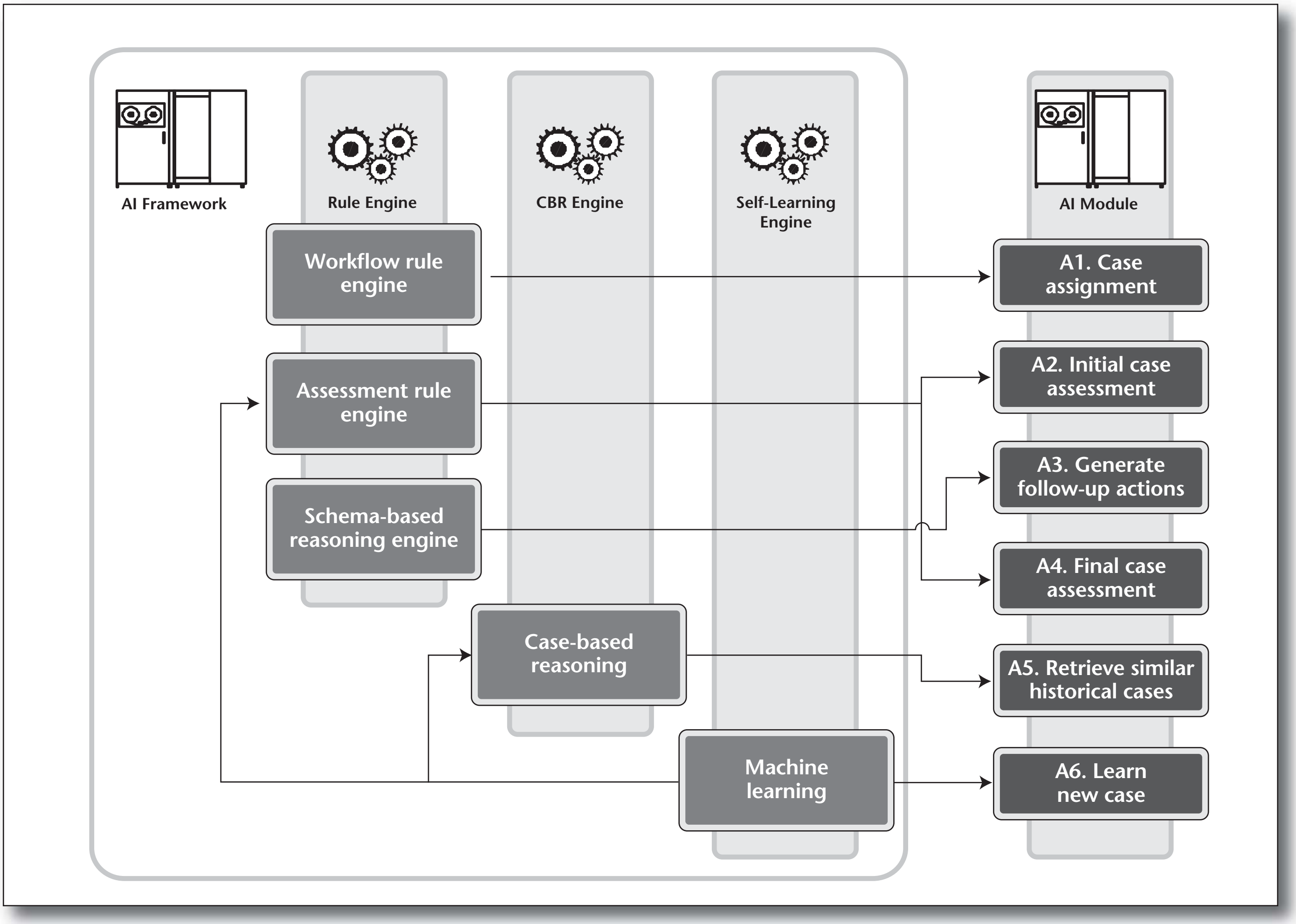

Figure 3. The Engines within the AI Framework.

Once a final decision has been made and the case is closed, the AI module will (A6) perform learning on the closed case. This involves indexing it into the case base (for future case retrieval) as well as decision trees (to learn current practices).

AI Architecture

To support the AI processing performed by the AI module, the AI framework makes use of several AI techniques, represented by five AI engines (figure 3).

Functionalities required by processes A1 to A4 are provided by rule-based technology. Case assignment (A1) is performed by a workflow rule engine. Initial (A2) and final case assessment (A4) are performed by an assessment rule engine. Follow-up actions (A3) are generated by a schema-based reasoning engine. A case-based reasoning (CBR) engine is used to retrieve similar historical cases (A5), and a self-learning engine is used to index and learn new cases (A6). Results from the self-learning engine feed back to the assessment rule engine as self-learned rules that represent current practices in discretionary decision making. Learning results also feed back to the CBR engine as newly indexed cases. To support these AI engines, the knowledge base consists of rules, schemas, cases, and decision trees.

Keeping It Manageable

Although there are over a hundred different types of forms, or application types, they are organized into only a few different categories or subsystems, such as the "right of abode," "certificate of entitlement," "birth, death, and marriage" (which includes adoption), "permits and visas," "travel pass system," "investigation," "nationality," "assistance to residents," and "electronic passport" subsystems.

To keep the AI development and deployment manageable, each subsystem has its own customized version of the AI module. The AI architecture (figure 3) is replicated for each subsystem. For example, the electronic passport subsystem has its own set of rules, schemas, cases, and decision trees.

Related Work

The AI work is built upon several different AI representations and reasoning algorithms—rules, schema-based reasoning, clustering, case-based reasoning, and decision trees.

Rules

The rule engine is similar to rules in traditional expert systems (Forgy and McDermott 1977, Buchanan and Shortliffe 1984). However, instead of heuristics or rules of thumb, the rules encode legislative knowledge (Gardner 1987). Each subsystem has its own rule base. The structure of the rule base was designed to facilitate ease of encoding expert knowledge on immigration-related legislation. A subsystem may have many different types of application forms. Each type of application has its rule agenda that defines which combination of rules or rule sets is applicable for a particular application type. The rule agenda is similar to other rule agendas[1, 2, 3, 4] except that its main purpose is to encode relationships among rules rather than just sequence. Beside the rule agenda, rules are also organized into rule sets (Quintus Prolog 2007, Jess 2007, CLIPS 2007). Each rule set represents one assessment criterion. Rules in a rule set represent how that criterion can be satisfied. Rules in the system operate in a forward-chaining manner.

Many government agencies around the world use rule engines to assist with decision making (Administrative Review Council 2004). For example, the Australian Department of Agriculture, Fisheries, and Forestry uses rule-based systems to make decisions on whether to permit or reject an import, whether to perform import inspections, and to determine what kind of tests to apply. The Australian Taxation Office also uses a number of rule-based systems to assist in determining which methods should be used in calculating taxes, benefits, and penalties. Customs uses expert systems to valuate imports, calculate customs taxes, and profile and select high-risk import or export transactions for scrutiny. The Department of Defence uses rule-based systems to calculate workers' compensation. The Department of Health and Ageing uses a rule-based system to check approved providers' compliances. The Department of Veterans' Affairs uses a rule-based system to support decision makers in determining veterans' entitlements.

In the United States, the Customs and Border Protection agency uses an expert system called the Automated Targeting System (ATS) (U.S. House of Representatives 1997, U.S. General Accounting Office 2004, U.S. Bureau of Customs and Border Protection 2005) to find suspicious cargo transactions and for antiterror work. ATS has more than 300 rules provided by field personnel, inspectors, and analysts in order to separate high-risk shipments from legitimate ones.

Schemas

Besides rules, the AI module uses schema-based reasoning (Turner and Turner 1991, Turner 1994) to represent procedural knowledge of actions and tasks that the case officers may take in the course of handling a case, for example, requests for verification of certain documents, letters of reference, and so on. Actions and tasks are triggered by rules. The schema encodes procedural knowledge of typical steps or actions taken by case officers in handling different kinds of cases.

Schema-based reasoning was also used in SAIRE (Odubiyi et al. 1997), a multiagent AI search engine to search Earth and space science data over the Internet. Chen and Lee (1992) explored how schema-based reasoning can identify fraud potentials exposed by an internal accounting control system.

Cases and Clustering

To provide decision support and precedent case retrieval, the AI module makes use of incremental (Ester et al. 1998) AI clustering (Fisher et al. 1993) with multivalued attributes (Ryu and Eick 1998) using k-means clustering algorithm.[5] AI clustering has been used successfully for many similar applications, such as QCS (query, cluster, summarize) (Dunlavy et al. 2006)—an information retrieval system that allows users to retrieve relevant documents separated into topic clusters with a single summary for each cluster. IBM Research (Campbell et al. 2006) developed a clustering system for indexing, analysis, and retrieval of videos.

The case-based reasoning (Kolodner 1993) engine makes use of AI clustering results to retrieve similar relevant cases to create recommendations and summaries. CBR is a popular approach to reuse previous experience to handle new situations. For example, PlayMaker (Allendoerfer and Weber 2004) is a CBR prototype that models how air traffic controllers handle traffic flow under severe weather or congestion. Xu (1996) used CBR to identify people who are "AIDS risky" to provide intervention and prevention. Esmaili et al. (1996) used CBR for computer intrusion detection.

Decision Trees

Finally, I use incremental decision trees (Mitchell 1997, Winston 1992, Utgoff 1989) to perform machine learning and rule generation (Quinlan 1987) to capture how case officers handle nonstandardcases. To enable decision trees to integrate back to the rule engines, each decision tree represents one assessment criterion, which is represented by a rule set in the rule engine.

In Australia, the Department of Family and

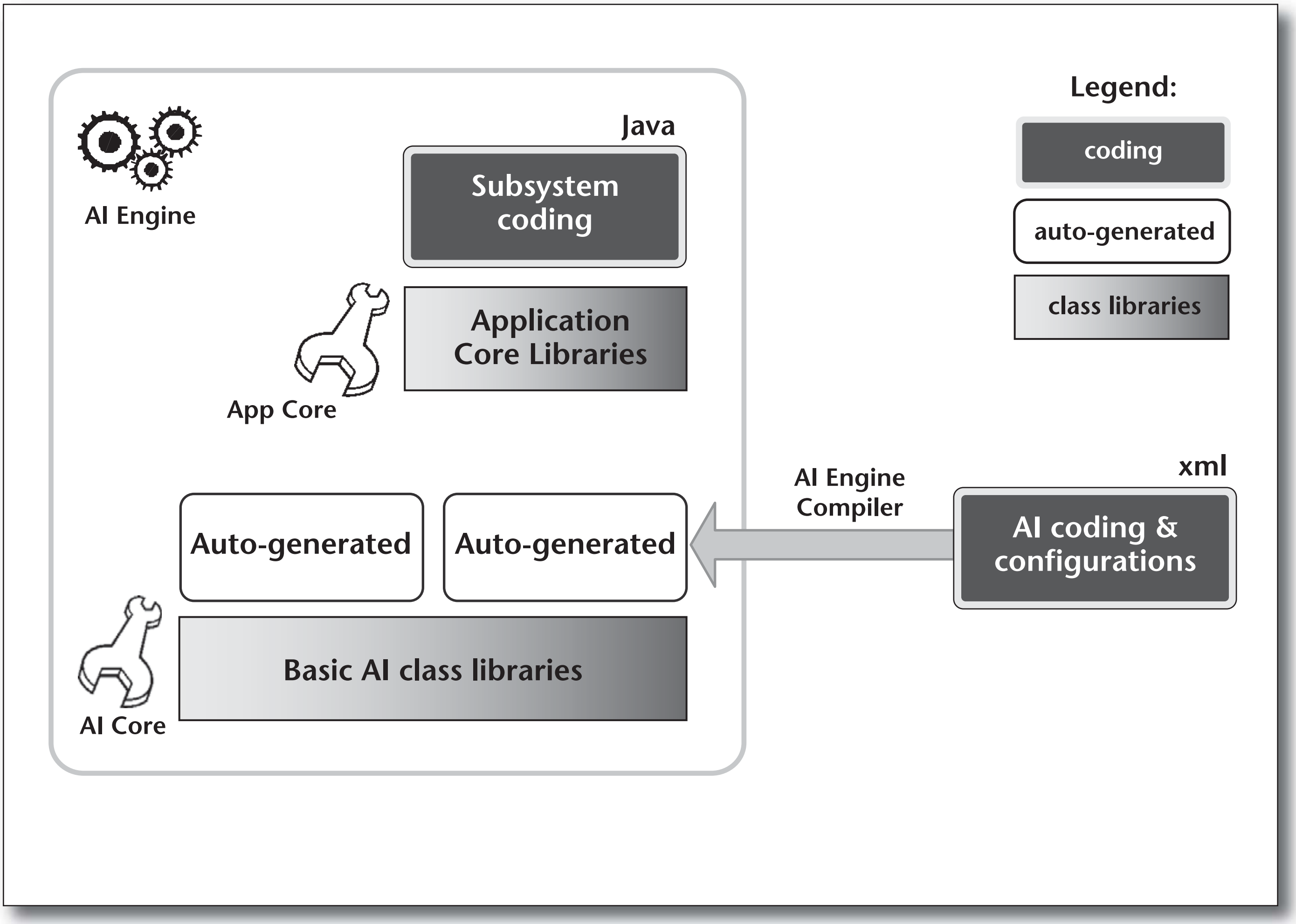

Figure 4. The Structure of each AI Engine.

Community Services has an "edge expert system" that uses decision trees to determine a citizen's likely entitlement to payments and services (Administrative Review Council 2004).

Application Description

The platform for the form-processing system is Java 2 Platform, Enterprise Edition. The AI module is therefore also Java-based and packaged and deployed as Java Enterprise Archive (EAR) files. For scalability, AI services are provided in a stateless manner and can be deployed on as many application servers as needed.

The front end to the AI software is a web-based thin client operated by immigration case officers. The layout and design of the web client is typical of other form-based systems. Each case officer has an inbox containing all the applications he or she has been assigned to handle. For each application, there are several screens to display personal information on the applicant, the details of the current application, related documents provided by the applicant and those sent to the applicant by the immigration agency, historical record on this applicant such as prior applications, other related information, and follow-up actions. Basically, anything related to an applicant, all his or her current and past applications and documents, is all consolidated in a conveniently accessible dashboard for the case officer to review.

The various AI features used by case officers operate in near real time. The other AI tasks that are not performance critical, such as case learning and rule generation, are done behind the scenes as background processes.

AI processing results are displayed on two key screens—an assessment screen and a decision support screen. The assessment screen displays results of the assessment rule engine as a list of violated rules and details of those rules, such as attributes and parameter values, as well as links to legal references related to that rule. Rules may include hard rules, soft rules, or self-learned rules. The assessment screen also contains a set of recommended follow-up action to take (generated by the schema-based reasoning engine), such as requesting additional documents or verifying the validity of certain information provided by the applicant. Actions already taken are also displayed.

The decision support screen is used to handle more complex, nonstandard cases. It displays a list of related precedent cases and their key attributes, generated by the CBR engine through AI clustering. In addition, the case officer can request the CBR engine to search for similar cases based on a selected subset of those attributes. The assessment results are also shown with reasons for approval or rejection.

Scalable AI Development

One of the key AI design objectives is that the development approach must be scalable so that it will be easy and convenient to extend the AI capabilities to cover the hundred different types of forms. Ease of maintenance is another important design objective to ensure that knowledge can be updated easily without any impact to any other deployed components.

The AI development approach created for this application is a nonintrusive XML-based approach in which all knowledge and configurations are coded using only resource-definition framework (RDF) or XML documents.[6, 7] Automatic code-generation techniques were then used to dynamically generate the actual AI engines as Java binaries, associated object-relational mappings, and database tables (see figure 4). This greatly shortens development time and minimizes potential coding errors (Tabet, Bhogaraju, and Ash 2000).

Other rule engines also use XML to encode rules.[8, 9, 10] This approach is used extensively for all the AI engines, not just the rule engines. Furthermore, the interface codes to the relevant domain objects that represent the application details are also autogenerated from RDF or XML and require no Java coding. In the past, interfacing an AI engine to an application has usually been very time consuming and error prone. To further simplify interfacing, results from AI processing are simply returned as an encapsulated result object.

Figure 4 shows the structure of an AI engine. Each of the five AI engines (figure 3) within each AI module of each subsystem follows a similar structure and development approach. At the bottom is the AI core—basic AI Java class libraries that represent AI algorithms and routines. On top of that are the subsystem-specific AI engine, object and relational mappings,[11] and database that are generated automatically from RDF or XML using the AI engine compiler.

For a rule engine, the RDF and XML documents describe the domain objects, rules, agendas, and rule sets. For CBR, the RDF and XML documents define the attribute vectors used for clustering. For decision tree learning, the RDF and XML documents define the decision trees and attributes within each tree. The crux of the AI development effort is in creating these RDF/XML documents. AI development is greatly simplified because, first, RDF and XML documents are easier to create (either directly or through a graphical user interface) than Java source code, and secondly, since Java binaries are created automatically, debugging time is eliminated. Nevertheless, there are still several thousand rules that need to be encoded, more than a hundred clustering vectors, and hundreds of decision trees.

The autogenerated AI engine requires a controllerto provide application-level APIs to control how the AI engine should operate and how results should be returned. The controller is provided by the app core—a set of application-specific class libraries. These Java libraries are shared by all subsystems. The only custom Java coding that is needed for each subsystem is the subsystem coding that defines behaviors and process flows specific to a particular subsystem. The amount of Java coding is very small and is needed only if the subsystem does not follow standard processing defined in the app core.

Although there are many subsystems and over a hundred types of application forms, the unique nonintrusive XML-based development approach makes the AI module very easy to customize and maintain. The generated AI engines represent AI services that are decoupled from other components within the application.

Uses of AI Technology

This section provides further details on how the different AI engines are used to streamline e-government forms processing.

The Assessment Rule Engine

The assessment rule engine is probably the most important as it encodes immigration-related legislative knowledge and guarantees that all applicable laws, regulations, and guidelines have been considered for each and every case. The key functions are (1) perform initial preliminary assessment to assist the workflow engine in case assignment as well as to guide information collection, and (2) perform final assessment to determine the application result.

There is one rule engine and hence one rule base per subsystem. Each subsystem may have many different types of application forms. To organize this large body of legislative knowledge, the rule base for each subsystem contains a separate rule agenda for each application type. The agenda determines which combination of rule sets is applicable for a particular application type. Rule sets contain all the rules related to determining the status of particular assessment criteria. For example, whether a person is a recognized citizen or not is one criterion in determining his or her right of abode. For this criterion, there are more than 30 different rules to help determine whether that criterion is satisfied or not. All those rules are stored in the "citizen" rule set and controlled by agendas that require this criterion in assessment of their associated application type.

Although most laws and regulations regarding immigration are relatively static, some of the guidelines do change from time to time. To facilitate user maintenance of rules without having to regenerate and republish a new rule engine each time, the rules are designed with parameter-driven capabilities. Parameter values can be edited by users with appropriate authority; the effect on related rules is instantaneous.

Beside established legislative knowledge, the rule engine also uses knowledge on discretionary decision making. This knowledge is taken from self-learned rules that are generated by the self-learning engine from observing how case officers handle nonstandard cases.

Before AI, assessments were done by sorting through and reviewing paper documents submitted by the applicant and using the case officer's own personal experience and knowledge of laws and guidelines. Time needed might be a few minutes for very simple cases to much longer for complicated cases. Some cases may take several days as the case officer may need to seek help and advice from other officers or superiors. With AI, assessment is done in less than 10 seconds for all cases regardless of complexity, while guaranteeing that all relevant legislation and guidelines, as well as all possible approval scenarios, are considered.

The Schema-Based Reasoning Engine

The schemas stored in the schema-based reasoning engine (Turner and Turner 1991) represent procedural knowledge in processing applications. It is used to generate tasks, checklists, and follow up actions for the case officer to perform.

It guides the case officer in collecting all necessary information and supplementary documents as well as printing documents and instructions. The engine is itself rule driven. This allows different sets of steps and actions to be proposed depending on the particulars of the application at hand and previous actions already taken.

Before AI, if there were unclear points in the application or if certain information needed to be verified, the case officer sent a letter to the applicant for additional supporting documents. After receiving the documents, the case officer analyzed the case once again and possibly requested more information from the applicant if needed. This cycle was time-consuming and stressful for the applicant as he or she may have needed to visit the agency several times before his or her application could be assessed. With AI, different scenarios are analyzed automatically at the same time, and a consolidated list is generated, thus minimizing the number of visits an applicant needs to make to the agency. Furthermore, letters to applicants are generated automatically. A task checklist is also provided to keep track of tasks so that nothing is overlooked.

The Case-Based Reasoning Engine

Straightforward cases are handled automatically by the assessment rule engine. But in the real world, there are many nonstandard cases that require more detailed analysis. This complicates and lengthens the assessment process. The CBR engine helps alleviate this situation by retrieving relevant closed cases from the case base to act as precedents or reference and indexing newly closed cases into the case base.

There is one CBR engine per application type since the attributes considered by different application types will be different. Each case is represented by a prioritized attribute vector that contains either data from the application form or results from the assessment rule engine. The objective of the CBR is to retrieve similar cases and use statistics to generate recommendations. The CBR engine also supports advanced features, such as multivalued attributes and incremental AI clustering (Ester et al. 1998).

The case officer may fine-tune the way the CBR engine retrieves relevant cases by selecting the assessment criteria that he or she feels are most important for the case at hand.

Before AI, it was very difficult if not impossible to locate past cases to use as reference or establish precedents. There was no easy way to search through microfiche or case folders, which were stored in physical archives. Case officers had to rely on their own memory or personal notes, or ask other case officers if they remembered handling similar cases before. Even if cases could be recalled, trying to retrieve them from microfiche or the archives took time. With AI, a precise list of all similar cases within a given time period can be retrieved within seconds, with all the details of the cases and the analysis results and comments from the case officers in charge.

The Self-Learning Engine

The self-learning engine captures discretionary decision-making knowledge that represents case officers' experience in handling special cases as well as their knowledge of assessment best practices and guidelines that change with the changing needs of society. The key functions are (1) incrementally learn and index new cases into decision trees, and (2) generate self-learned rules from the decision trees and integrate them into the assessment rule engine

For the same reasons as the CBR engine, there is one self-learning engine per application type. However, each engine may contain many decision trees. Each decision tree represents knowledge related to one assessment criterion. These are the same assessment criteria used by the rule engines as well as the CBR engines. The decision tree is either constructed from prioritized data from the application form or retrieved from results from the assessment rule engine. This engine also supports advanced features such as incremental learning.

The self-learned rules generated from each decision tree are used to determine whether an assessment criterion was fulfilled or not. Hence rules generated by the self-learning engine are well integrated with the assessment rule engine and directly contribute to the assessment result.

Before AI, discretionary decision making was based on practices and guidelines that were discussed and shared verbally among case officers. Each case officer kept a personal notebook of these guidelines and practices as reference. The performance of a case officer depended greatly on his or her knowledge of these practices and guidelines and his or her personal experience in handling similar cases. There was no easy way to share this type of knowledge efficiently before. With AI learning, patterns in discretionary decision making are extracted and codified as rules so that the current practices can be shared and used regardless of the experiences of the case officers.

Application Use and Payoff

The AI module was deployed to production in December 2006. Starting in early February 2007 it began to process each and every electronic passport application. Rollout for the remaining subsystems is scheduled throughout 2007 and 2008. So far, several hundred immigration case officers have been trained on the system.

Evaluation Results

Prior to deployment, extensive unit, integration, and stress testing was performed. After that, the system went through 2 months of user testing and 2 months in a production environment before the official launch in February 2007. Obviously, this type of AI system must return correct results 100 percent of the time and be fast (within seconds) and stable. General feedback and results from user evaluation included a number of conclusions. First, for subsystems with a large volume of applications, automatic assessment with AI rules is the only way to improve efficiency. Second, the ability to automatically find precedent cases is very important and highly useful for decision support. This is too hard to do with the old microfiche system. Third, self-learning is also very important for certain application types because rules can be too complex to code manually. Fourth, automatically consolidating all information related to a particular case into a dashboard was found to be very useful. The old approach of manually sorting through paper documents and records was too time consuming and error prone. Finally, the ability to automatically propose follow-up actions and to automatically generate notification letters and minutes was also found to be very useful and a major time savings.

Key payoffs include improved quality of service and assessment, increased productivity, improved agility, increased capacity for growth, and economic savings.

Quality of service is the number one priority for this immigration agency. Year after year, it receives numerous awards and recognitions for outstanding quality of service to citizens and visitors (Hong Kong Government 2007a). The use of AI to streamline processing workflow enhances the quality of service by reducing turnaround time (Hong Kong Government 2007b). For example, time to get an entry permit for employment can be shortened by 3 to 5 days, whereas a search of birth, death, or marriage records can be reduced to several minutes. One-stop service can be possible for some applications. Second, the use of AI provides a more comprehensive and thorough assessment of each case so that follow-up tasks are consolidated, minimizing the number of documents the applicant must provide and the number of visits to the agency.

In the past, *assessment quality* depended on the experience and knowledge of case officers. Time was needed to think through numerous intricate and complex laws and regulations for each type of application. Manual assessment was time consuming and error prone. With AI, all relevant laws, regulations, and guidelines are considered at all times within seconds, guaranteeing that nothing is overlooked and eliminating any potential for errors.

Increased productivity was another key payoff. For complicated cases, case officers need time to sort out all the information provided by the applicant as well as run through different approval scenarios. This can be time consuming and may require discussions with other case officers to clarify fine details of legislation. With AI, applications are assessed under all possible scenarios at the same time and within seconds. In addition, locating historical cases from microfiche or folders from physical archives was previously very time consuming. Using AI, relevant cases are automatically retrieved without any effort from the case officer. Case officers can then focus on using their expertise more effectively for decision making.

The system also offered improved agility. Because the AI module is parameter driven, any urgent change in guidelines and policies can be made instantly without any change to software. With self-learning capabilities, the AI module automatically adapts itself to changing practices and guidelines. Hence the agency becomes more agile in terms of its knowledge management capabilities.

Another payoff was in capacity for growth. In the long term, the AI module will allow the agency to cope with continuously increasing workloads to support the city's economic growth.

A final key payoff was economic savings. The agency estimates that the application processing system will save the government more than US$16 million annually (Hong Kong Government 2004). Efficiencies provided by the AI module represent not only cost savings for the government but also substantial savings in community cost in reduced waiting and turnaround time for citizens and visitors.

Application Development and Deployment

The design and development of the application processing system began in early 2005 with the AI work starting in mid-2005. The project prime contractor is NCSI,[12] a wholly owned subsidiary of NCS, a leading IT solutions provider headquartered in Singapore with several thousand IT professionals worldwide. AI technology for the project was provided by CityU Professional Services Limited, a nonprofit subsidiary of the City University of Hong Kong.

The total IT team for the entire project consists of roughly 200 programmers, system analysts, and consultants from several IT vendors and system integrators from around the world. In addition, roughly another 60 officers and managers from the user side are dedicated to this project.

The AI design and development team consists of roughly 10 knowledge engineers and AI developers. AI development was simplified with extensive support from the user side in providing knowledge in a form that was readily convertible into rules for the rule engine.

For a system as complex as this, integration, robustness, and scalability were major concerns in the design of the AI module. To minimize integration issues, the AI module was designed to be decoupled from other components using well-defined interfaces. Robustness is handled by designing the AI module to be deployable per subsystem or even per application type if needed. Any fault in one subsystem or application type will not affect others. In addition, all internal databases used by the AI module have redundancy to improve robustness and performance. Scalability is handled by designing the AI module to provide AI services in a stateless manner. If workload increases, all that is needed is simply to add more application servers. This distributed design also allows the application to switch over to another AI server if one fails.

Deployment

AI deployment is prioritized based on subsystems and application types with the "electronic passport" (Xinhua News Agency 2007, Hong Kong Government 2007c) and "birth, death, and marriage" subsystems to be the first to be deployed.

The first version of the assessment rule engine and schema-based reasoning engine was released in mid-January 2006. This was followed by the CBR engine in mid-February 2006 and the self-learning engine at the end of March 2006. Since then, the systems have been undergoing extensive testing. In parallel, the engines were customized for different subsystems and application types as well as fine-tuning features and performances.

User testing began in September 2006 with the first rollout to production in December 2006. Subsequent subsystems are scheduled to be deployed throughout 2007 and 2008.

Maintenance

Just as with any other mission-critical software, there will inevitably be changes and upgrades to the AI module after deployment to reflect legislative or operational changes for the agency. The design of the AI architecture is such that these types of changes are very easy to do.

First, all knowledge-related changes can be done without any Java coding simply by updating the RDF and XML documents and configuration files. Binaries and databases are then generated automatically by the engine compilers. Second, the behavior of the rule engines are parameter driven and under user control to reduce the need for code change. Packaging the AI module as a decoupled component from the other parts of the system helps further reduce maintenance and integration needs.

For support, the IT team of the prime contractor, NCSI, provides front-line technical and end-user support while CityU provides additional assistance on the AI technologies when needed.

Conclusion

This article provided an overview of how various AI techniques were used to provide highly intelligent and accurate case assessment capabilities to an e-government form-processing system. AI streamlines processes and results in higher quality and faster service to citizens and visitors. In addition, valuable domain knowledge and expertise related to immigration laws, regulations, and guidelines are now quantified, coded, and preserved within the agency for use in this and other systems. The AI work makes use of several innovative techniques, such as nonintrusive RDF and XML coding, and integrated rule, schema-based, case-based, and self-learning engines as well as incremental clustering and learning. This may be the first time any immigration agency in the world is using AI for automated assessment and in such a large and broad scale of deployment.

Notes

1. See the CLIPS website (www.ghg.net/clips/CLIPS.html).

2. See the Quintus Prolog website (www.sics.se/isl/quintuswww/site/flex.html)

3. JBoss Rules (www.jboss.com/products/rules).

4. See Jess—the Rule Engine for Java Platform (herzberg.ca.sandia.gov/jess/).

5. See the Wolfram MathWorld website (mathworld.wolfram.com/K-MeansClusteringAlgorithm.html).

6. The resource description framework (RDF) is available at www.w3.org/RDF.

7. The RDF/XML Syntax Specification (Revised), W3C Recommendation 10 February 2004 is available at www.w3.org/TR/rdf-syntax-grammar.

8. See the Drools website (drools.org/).

9. See A Java Deductive Reasoning Engine for the Web (jDrew) (www.jdrew.org).

10. See the Mandarax Project (mandarax.sourceforge.net).

11. See the Hibernate website (www.hibernate.org).

12. See the NCS website (www.ncs.com.sg)

References

Allendoerfer, K. R., and Weber, R. 2004. PlayMaker: An Application of Case-Based Reasoning to Air Traffic Control Plays. In *Proceedings of the Seventh European Conference on Case-based Reasoning (ECCBR),* 476–488. Berlin: Springer.

Administrative Review Council, Government of Australia. 2004. Automated Assistance in Administrative Decision Making, Administrative Review Council, Report to the Attorney-General, Report No. 46, November 2004. Canberra, ACT: Government of Australia (www.ag.gov.au/agd/www/archome.nsf/Page/RWPD0BD2AED121A15A8CA256F5E00022956).

Buchanan, B. G., and Shortliffe, E. H., eds. 1984. *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Reading, MA: Addison-Wesley.

Campbell, M.; Haubold, A.; Ebadollahi, S.; Naphade, M. R.; Natsev, A.; Seidl, J.; Smith, J. R.; Tei, J., and Xie, L. 2006. IBM Research TRECVID-2006 Video Retrieval System. In *Proceedings of the NIST TRECVID-2006 Workshop*. Gaithersburg, MD: National Institute of Standards and Technology.

Captiva Software Corporation. 2002. Swiss Tax Office Processing More Returns for Less. Hopkinton, MA: EMC Corporation (www.captivasoftware.com/products/casestudies/casestudies_view.asp?wcs_id=55).

Chen, K. T., and Lee, R. M. 1992. Schematic Evaluation of Internal Accounting Control Systems. EURIDIS Research Monograph, RM-1992-08-1. Erasmus University, Rotterdam, The Netherlands, August 1992.

Dunlavy, D. M.; O'Leary, D. P.; Conroy, J. M.; and Schlesinger, J. D. 2006. QCS: A System for Querying, Clustering, and Summarizing Documents, Sandia Technical Report, Sandia National Laboratories, Albuquerque, NM, July.

Esmaili, M.; Balachandran, B.; Safavi-Naini, R. and Pieprzyk, J. 1996. Case-Based Reasoning for Intrusion Detection. Paper presented at the Computer Security Applications Conference, San Diego, CA, 9–13 December.

Ester, M.; Kriegel, H. P.; Sander, J.; Wimmer, M. and Xu, X. 1998. Incremental Clustering for Mining in a Data Warehousing Environment. In *Proceedings of the 24th International Conference on Very Large Data Bases (VLDB).* San Francisco: Morgan Kaufmann Publishers.

Fisher, D.; Xu, L.; Carnes, J.; Reich, Y.; Fenves, S.; Chen, J.; Shiavi, R.; Biswas, G.; and Weinberg, J. 1993. Applying AI Clustering to Engineering Tasks. *IEEE Expert* 8(2): 51–60.

Forgy, C., and McDermott, J. P. 1977. OPS, A Domain-Independent Production System Language. In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence (IJCAI-1977)*, 933–939. Los Altos, CA: William Kaufmann, Inc.

Gardner, A. 1987. *An Artificial Intelligence Approach to Legal Reasoning*. Cambridge MA: The MIT Press.

Hong Kong Government. Special Administrative Region of the People's Republic of China. 2004. Implementation of Phase III of the Updated Information Systems Strategy for the Immigration Department, FCR(2004–05)10, 14 May (www.legco.gov.hk/yr03-04/english/fc/fc/papers/f04-10e.pdf).

Hong Kong Government. Special Administrative Region of the People's Republic of China. 2005. Immigration Department Annual Report 2004–2005. Hong Kong: Government of the People's Republic of China (www.immd.gov.hk/a_report_04-05/htm_en/ch05/url/frame_020.htm).

Hong Kong Government. Special Administrative Region of the People's Republic of China. 2007a. Recognition of the Quality Service of Immigration Department, Immigration Department. Hong Kong: Government of the People's Republic of China (www.immd.gov.hk/ehtml/rqs.htm).

Hong Kong Government. Special Administrative Region of the People's Republic of China. 2007b. Performance Pledge, Immigration Department. Hong Kong: Government of the People's Republic of China (www.immd.gov.hk/ehtml/pledge.htm).

Hong Kong Government. Special Administrative Region of the People's Republic of China. 2007c. HKSAR Electronic Passport (e-Passport) and Electronic Document of Identity for Visa Purposes, Immigration Department. Hong Kong: Government of the People's Republic of China (www.immd.gov.hk/ehtml/eppt_edi.htm).

Hong Kong Trade Development Council. 2006. HKSAR Immigration Department: Aiming at Security and Quality Service, May (ict.tdctrade.com/suc-e513.htm).

Kolodner, J. 1993. *Case-Based Reasoning*. San Mateo, CA: Morgan Kaufmann Publishers.

Mitchell, T. 1997. *Decision Tree Learning, Machine Learning,* 52–78. New York: McGraw-Hill.

Odubiyi, J. B.; Kocur, D. J.; Weinstein, S. M.; Wakim, N.; Srivastava, S.; Gokey, C. and Graham, J. 1997. SAIRE—A Scalable Agent-Based Information Retrieval Engine. In *Proceedings of the First International Conference on Autonomous Agents*, 292–299. New York: Association for Computing Machinery.

Questex Media. 2006. The HK Immigration Department's Continuing Drive for Automation. *Enterprise Innovation* 15 August (www.enterpriseinnovation.net/perspectives.php?cat1=4&id=816).

Quinlan, J. R. 1987. Generating Production Rules from Decision Trees. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence (IJCAI 1987),* 304–307. San Francisco: Morgan Kaufmann Publishers.

Ryu, T. W., and Eick, C. F. 1998. Similarity Measures for Multivalued Attributes for Database Clustering, In *Proceedings of the Conference on Smart Engineering System Design: Neural Networks, Fuzzy Logic, Evolutionary Programming, Data Mining and Rough Sets*. New York: ASME Press.

Tabet, S.; Bhogaraju, P.; and Ash, D. 2000. Using XML as a Language Interface for AI Applications. In *Advances in Artificial Intelligence, PRICAI 2000 Workshop Reader, Four Workshops held at PRICAI 2000, Lecture Notes in Computer Science 2112,* 103–110. Berlin: Springer-Verlag.

Turner, E. H.; and Turner, R. M. 1991. A Schema-based Approach to Cooperative Problem Solving with Autonomous Underwater Vehicles. In *Proceedings of the IEEE Oceanic Engineering Society OCEANS'91*. Piscataway, NJ: Institute of Electrical and Electronic Engineers.

Turner, R. M. 1994. *Adaptive Reasoning for Real-World Problems: A Schema-Based Approach*. Hillsdale, NJ: Lawrence Erlbaum Associates.

U.S. Bureau of Customs and Border Protection. 2005. Targeting Center the Brains behind Anti-Terror Efforts. *Customs and Border Protection Today,* (November/December) (www.cbp.gov/xp/CustomsToday/2005/nov_dec/targeting.xml).

U.S. General Accounting Office. 2004. Homeland Security: Summary of Challenges Faced in Targeting Oceangoing Cargo Containers for Inspection, March 2004, GAO-04-557T. Washington, D.C: U.S. Government Printing Office (www.gao.gov/htext/d04557t.html).

U.S. House of Representatives. Subcommittee on Immigration and Claims, Committee on the Judiciary. 1997. Border Security and Deterring Illegal Entry into the United States, April 1997. Washington, D.C.: U.S. Government Printing Office. (commdocs.house.gov/committees/judiciary/hju43664.000/hju43664_0f.htm).

Utgoff, P. E. 1989. Incremental Induction of Decision Tree. *Machine Learning* 4: 161–186.

VertMarkets IT Group. 2003. Australian Government Agency Improves Immigration Records Keeping, Lowers Operating Costs. *Enterprise Content Management News and Solutions,* 11 December (www.ecmconnection.com/Content/news/article.asp?Bucket=Article&DocID={91F02017-2FF4-4DC6-BEAD-BEA3B9B44620).

VertMarkets IT Group. 2002. U.S. Census Demands Reliable Imaging Solution. *Enterprise Content Management News and Solutions*, 9 April (www.ecmconnection.com-Content/news/article.asp?Bucket=Article&DocID=%7B954C8253-9B23-4BF3-B3BF-A3CD94FF45C6%7D).

Winston, P. 1992. *Learning by Building Identification Trees, Artificial Intelligence,* 423–442. Reading, MA: Addison-Wesley.

Xinhua News Agency. 2007. Hong Kong to Issue e-Passport, 1 January 2007. Hong Kong: Government of the People's Republic of China (english.gov.cn/2007-01/01/content_485516.htm).

Xu, L. D. 1996. An Integrated Rule- and Case-Based Reasoning to AIDS Initial Assessment. *International Journal of Bio-medical Computing* 40(3): 197–207.

Andy Chun is an associate professor in the Department of Computer Science at the City University of Hong Kong. His research interests include web technologies; search engine optimization; scheduling, rostering, optimization; business intelligence and data mining; and distributed architectures. He received a B.S. from the Illinois Institute of Technology and an M.S. and a Ph.D. in electrical engineering from the University of Illinois at Urbana-Champaign. His e-mail address is andy.chun@cityu.edu.hk.