Automating the Underwriting of Insurance Applications

Kareem S. Aggour, William Cheetham

General Electric Global Research
One Research Circle
Niskayuna, NY 12309
aggour@research.ge.com, cheetham@research.ge.com

Abstract

An end-to-end system was created at Genworth Financial to automate the underwriting of Long Term Care (LTC) and Life Insurance applications. Relying heavily on Artificial Intelligence techniques, the system has been in production since December 2002 and today completely automates the underwriting of 19.2% of the LTC applications. A fuzzy logic rules engine encodes the underwriter guidelines and an evolutionary algorithm optimizes the engine's performance. Finally, a natural language parser is used to improve the coverage of the underwriting system.

Introduction

With over 130 years of history, 15 million customers, $98 billion in assets, and $11 billion in annual sales, Genworth Financial (GNW) is one of the world's oldest and largest insurance providers. GNW is committed to providing financial protection to its customers, their families, and their businesses. This is accomplished through a diverse set of products, including Long Term Care, Term Life, Dental, Disability, and Mortgage Insurance. Long Term Care (LTC) Insurance is used to cover significant medical costs, such as home nursing care, to protect the policyholder's assets through illness and old age. Term Life Insurance provides benefits to the living upon the death of the insured. This paper focuses on the automation of the LTC underwriting process, but much of the material applies to Term Life underwriting as well.

As GNW receives LTC insurance applications, an individual referred to as an underwriter reviews each to determine if the applicant should be approved for coverage. Based on the applicant's medical history, the underwriter assigns the applicant to a discrete risk category, or declines the applicant altogether. The risk category dictates the premium to be paid for the insurance, making appropriate placement critical. Underestimating the risk would result in the applicant not paying enough to cover the financial risk incurred insuring that individual. Overestimating the risk would result in GNW not being price competitive, and losing customers. Prior to this automation effort, this crucial underwriting process was entirely manual.

GNW chose to automate this process to improve consistency and reduce the number of defects. For legal reasons the decision-making process had to remain transparent, however, constraining the technologies that were used.

Manual Underwriting Process

The LTC underwriting process begins when a paper application (APP) is completed by hand, faxed to GNW, and then scanned into an electronic data warehouse. Underwriters located throughout the country view these scanned documents online, and then rate the risk of insuring each person. If the underwriter has any concerns, he can request additional information from the applicant via a Phone Health Interview (PHI) and/or a Face-to-Face (F2F) interview, resulting in the submission of additional paper forms. At any time, an underwriter can also request an Attending Physician Summary (APS)—a copy of the applicant's medical history from their primary physician. Before the automation of the underwriting process, volumes of these documents were ordered extraneously, providing no value at a great cost of time and money. One benefit of automation was reducing this waste.

An underwriter can make a decision at any point they feel they have sufficient information. If they have any questions or concerns, they can refer cases to a senior underwriter. Once a decision is made, the applicant is notified by mail. To evaluate the quality of the decisions produced, a percentage of the cases are randomly audited on a monthly basis. Figure 1 shows the manual process.

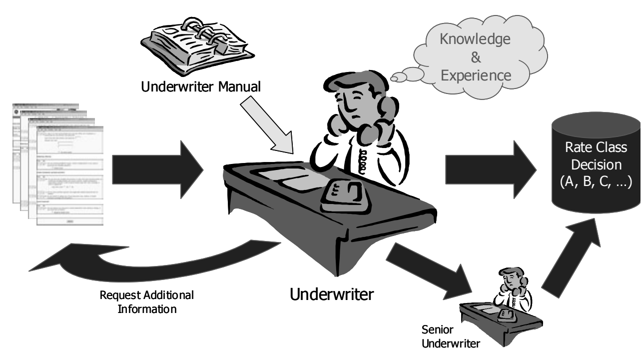
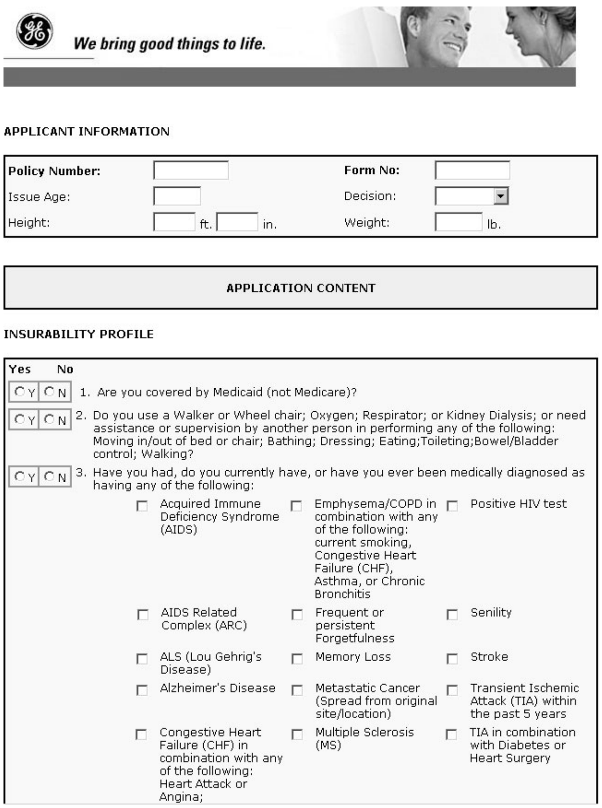


Figure 1: Manual Underwriting Process

Underwriters make decisions following guidelines specified in an underwriter manual. They also rely upon extensive medical knowledge and personal experience when underwriting cases. The reliance upon their own experience and judgment causes inconsistency across the underwriters, resulting in inaccurate rate classifications. This use of personal knowledge and experience to make decisions also made this a difficult problem to automate.


We bring good things to life.

APPLICANT INFORMATION

Policy Number: Form No:
Issue Age: Decision:
Height: ft. in. Weight: lb.

APPLICATION CONTENT

INSURABILITY PROFILE

Yes No

1. Are you covered by Medicaid (not Medicare)?

2. Do you use a Walker or Wheel chair; Oxygen; Respirator; or Kidney Dialysis; or need assistance or supervision by another person in performing any of the following: Moving in/out of bed or chair; Bathing; Dressing; Eating;Toileting;Bowel/Bladder control; Walking?

3. Have you had, do you currently have, or have you ever been medically diagnosed as having any of the following:

- Acquired Immune Deficiency Syndrome (AIDS)
- Emphysema/COPD in combination with any of the following: current smoking, Congestive Heart Failure (CHF), Asthma, or Chronic Bronchitis
- Positive HIV test
- AIDS Related Complex (ARC)
- Frequent or persistent Forgetfulness
- Senility
- ALS (Lou Gehrig's Disease)
- Memory Loss
- Stroke
- Alzheimer's Disease
- Metastatic Cancer (Spread from original site/location)
- Transient Ischemic Attack (TIA) within the past 5 years
- Congestive Heart Failure (CHF) in combination with any of the following: Heart Attack or Angina;
- Multiple Sclerosis (MS)
- TIA in combination with Diabetes or Heart Surgery

Figure 2: Part 1 of APP Summarization Form

Prior Art

The GNW system shares features similar to other automated underwriting applications, including the Countrywide Loan Underwriting Expert System (CLUES) described in (Talebzadeh, Mandutianu, and Winner 1994) and Desktop Underwriter (DU) described in (McDonald, Pepe, et al. 1997).

Both CLUES and DU used AI techniques to automate a manual mortgage loan underwriting process that faced similar problems to the manual medical insurance underwriting process, including a high case volume and subjective human reasoning impacting the accuracy of decisions. Each project—CLUES, DU, and GNW's system, required that decisions be explainable (ruling out black-box techniques) and the system be easily modifiable. Therefore, all three efforts opted to use rule-based expert systems at their core.

The next section describes the new automated process. The use of AI technology and the surrounding system are then presented. Benefits of the new system are provided, followed by details on the system development, deployment, and maintenance. Finally, some conclusions and future work are presented.

Automated Underwriting Process

In automating the underwriting process, Artificial Intelligence techniques were used to codify the underwriter rules. These rules were then incorporated into a new, automated end-to-end rule-based system (Chisholm 2004). A Fuzzy Logic Rules Engine (FLRE) was designed and developed to codify the underwriter rules (Jang, Sun, and Mizutani 1997); this became the 'digital underwriter' in the new process. This digital underwriter is able to determine if an application should be sent to a human underwriter for review, allowing the automated process to be deployed without worrying about every possible case variation. This enabled a staged rollout of functionality, shortening the time that was needed for the FLRE to provide value to GNW.

The progression of this system through three generations of development and deployment is described below.

First Generation

The first generation of the end-to-end system focused on the simplest subset of cases—applications with no medical impairments. The new process begins with a team of medical summarizers digitizing the scanned APPs. The summarizers view the scanned applications online and fill in Web-based forms to digitize them. A page from the APP summarization form is shown in Figure 2. Next, the digital application is passed through an instance of the FLRE (referred to as the APP-FLRE). The APP-FLRE makes three decisions:

1. In what rate class to place the applicant,
2. Whether or not to order additional information, and
3. Whether or not to send the case to a human underwriter for review (i.e., reverting to the manual process).

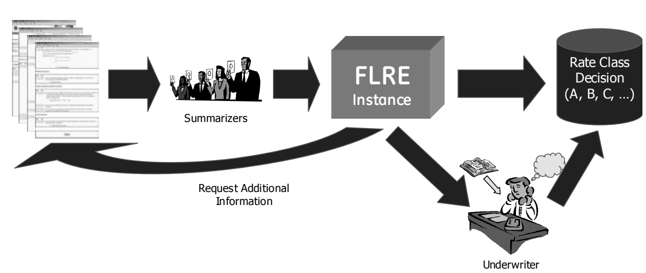


Figure 3: Automated Underwriting Process

If additional information is requested from the applicant, it is also digitized on arrival. The new content is then passed through a separate instance of the FLRE, using different rule sets but making the same three decisions. This new decision process is presented in Figure 3.

With multiple decision engines, more than one rate class decision may be made for a single applicant. The lowest rate class (i.e., highest premium) always takes precedence across all of the engines that may be invoked. For example, if an applicant has both an APP and a PHI, the lower decision is used.

If any of the engines decide a case should be sent to a human underwriter for review, that decision will be honored. Cases can be diverted back into the manual process any time an engine is unable to make a definitive decision. If an automated decision is made, a new notification system automatically mails a letter to the applicant with the decision.

An interesting parallel between CLUES and GNW is that both allow the engine to either make final decisions or decide to send an application back to a human underwriter for review. The CLUES team also found this to be a very effective means of improving decision throughput and accuracy, without requiring that the system be capable of completely replacing the human process on the first day of its rollout.

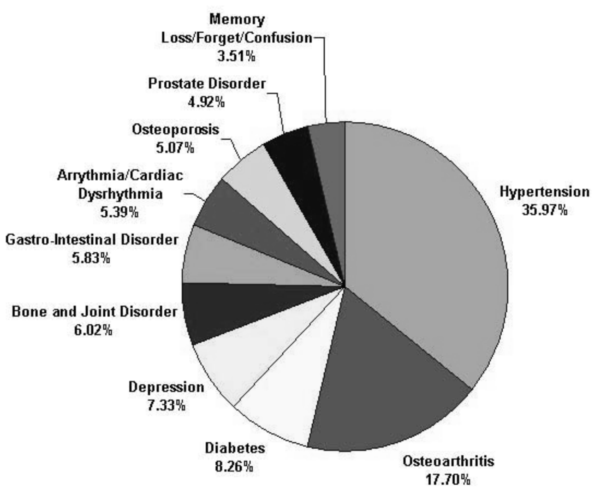


Figure 4: Relative Frequency of Impairments

Second Generation

The second generation of the system covered two major impairments. Statistics on the frequency of impairments in applications from the past seven years were obtained to drive the specific impairment selection. Figure 4 shows these relative frequencies.

The second generation of engines handled APS's with two of the most common medical impairments: Hypertension (HTN) and Diabetes Mellitus (DM). HTN was chosen because it is the most common impairment seen on applications. DM was chosen because it is also quite common and has one of the highest average claims costs. The coverage of these impairments required new Web forms for the summarizers to enter information about the impairments, new rules to determine rate classes from this information, and new rules to determine when applications with these impairments could be automated.

If an APS has been ordered, the medical summarizers review it, determine the applicant's impairments, and then complete the appropriate summarization forms. Separate FLRE instances are invoked as needed.

Third Generation

The third generation focused on three areas:

- Increasing the set of impairments covered,
- Increasing the number of applications that can be automated by adding natural language processing, and
- Assisting the underwriter when an application cannot be fully automated.

Two additional impairments were covered by the third generation of the system. Osteoarthritis (OA), the next most frequent impairment, was selected. Osteoporosis (OP) is closely related to OA, so this impairment was also covered.

Natural Language Processing

After Gen 2, a significant percentage of the applications containing impairments covered by the rules engines still could not be automated. The primary reason for this was the input from the summarizers occasionally contained free text that required review by an underwriter. Usually this free text does not affect the rate class decision, so if text entries could be interpreted and classified as benign, the level of automation could be increased. Example benign text includes "annual physical," "routine visit, everything normal," and "cholesterol check."

Classifying critical text as benign (i.e., false positives) is not acceptable, however it is acceptable to have errors where benign text is classified as needing review (i.e., false negatives). The latter type of errors result in underwriters performing the same tasks they currently do.

A natural language parser (Jurafsky and Martin 2000) was constructed to determine if the text entered by the summarizers is benign. A grammar was constructed for benign text and lists were created for:

- Noise words and in-phrase characters (Noise)
- Phrase separators (Separator)
- Benign words or synonyms (Benign)
- Dates in various formats (Date)

The current grammar for benign text is:

BenignText:

BenignPhrase [Separator [BenignPhrase]]*

BenignPhrase:

[Noise]* [Benign [Noise]* [Date [Noise]*]

Table 1: Natural Language Parser Accuracy

Version	False Benign	False Assist	True Benign	True Assist
Basic grammar	1.15	62.54	37.46	98.85
Dates parsed	1.15	62.35	37.65	98.85
Improved lists	0.60	38.08	61.92	99.40
Remove in-phrase characters	0.60	32.56	67.44	99.40
Match longest first	0.83	0.00	100	99.17
? not a separator	0.00	0.00	100	100

A training set was used with 160,408 entries, 70.4% of which were benign. A list of every unique word in the text was created, and each word was manually classified as benign or not. The evolution of the grammar above is shown in Table 1. A basic grammar excluding dates, noise words, and in-phrase characters was developed first. The accuracy of this grammar on the training set is shown in the first row of Table 1. The first column represents the percent of text phrases that are not benign, but were labeled as benign. These are the most significant classification errors. True benign is the percent of benign phrases that are correctly classified as benign. The larger the true benign, the greater the benefit of the natural language processing feature. A second version of the grammar added parsing multiple date formats, slightly increasing the true benign percentage, as shown in the second row of Table 1.

An expanded list of benign terms, which included synonyms and phrases, was then created. This greatly improved the true benign and reduced the false benign rates, as shown in the third row of Table 1. To improve the results further, characters such as the dash were treated specially. Next, the parser was modified so that longer phrases had priority over shorter phrases or single words. The true benign rate greatly improved at the expense of a small increase in the false benign rate, as shown in row five of Table 1. Finally, question marks were being used as indicators of uncertainty by the summarizers, instead of being at the end of sentences that are questions. Not counting the question mark as a separator produced the final accuracy found in the last row of Table 1.

After the parser was created, it was tested on a sample population of 36,635 benign and non-benign phrases. The result from this test set was also 0.00% false benign and 100% true benign. One reason for these surprisingly good results is the same summarizers were used to produce the training and test data. It is possible the accuracy would decrease if different people created the text phrases.

Some simple non-AI techniques were also used to limit the FLRE cases sent to the underwriter due to free text. This included summarizer training on how and when to enter free text and modifying the entry forms so that common comments could be selected with drop down lists, check boxes, or other non-text based methods. New rules were created for these new data elements.

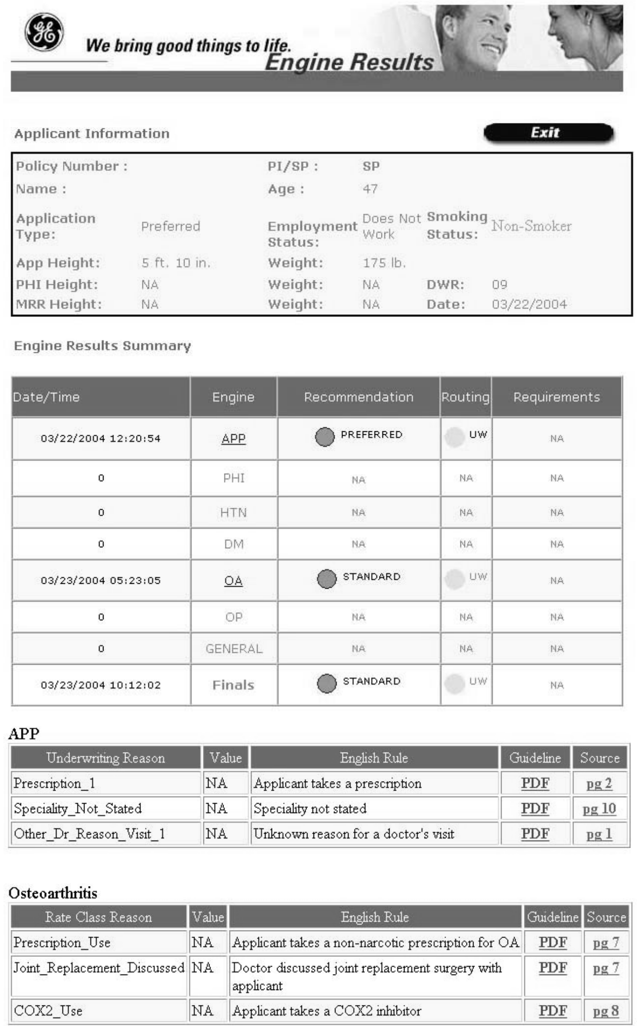


Figure 5: Underwriter Assist Screen

Underwriter Assist

The third focus of Gen 3 was to develop a way to help the underwriter when an application could not be placed by the FLREs. This occurred in about eighty percent of the applications. In the first two generations, if an application was sent to an underwriter, he had to start on the application from scratch with no visibility into what the FLREs had suggested. For example, if six FLREs had proposed a rate class and one said the underwriter needed to be involved, then the six rate class decisions would all be ignored.

The underwriters' productivity could be improved if the system could propose a rate class for each portion of the application where it was confident in its decision. If an FLRE was not confident, then it should highlight the reason for its lack of confidence. Figure 5 shows an early prototype of an underwriter assist screen. The top section has applicant information such as name, age, height and weight. The next section has a summary of each FLRE result, with one row for each engine. In this example, only the application and the OA-FLRE applied to the applicant, as he did not have any other impairment. The engine result summary has five columns:

1. The date and time the engine was run
2. The name of the specific engine
3. The recommended rate class
4. Where to route the application (UW to send to underwriter)
5. Requirements for additional tests needed

The APP section gives details about the APP-FLRE rules that caused the rate class recommendation and routing. In this example, there was an unknown reason for a doctor visit that needed to be obtained. The underwriter can click on the PDF Guideline for a complete description of the rule invoked. The original information sent to the summarizer that applied to this rule can be seen by clicking on the pages listed in the 'Source' column.

The OA-FLRE sent this application to the underwriter because the applicant's doctor discussed joint replacement surgery with the applicant. The underwriter should therefore investigate the severity of the need for surgery, which would significantly impact the applicant's rate class. This interface provides the underwriter with the ability to get an immediate assessment of the applicant and focus his attention on the problem areas instead of having to review the entire application.

Use of AI Technology

Fuzzy logic rules are used to encode underwriting standards. Fuzzy logic is a superset of conventional Boolean (True/False or 1/0) logic, allowing truth-values to be equal to any real number in the interval [0,1], with intermediate values denoting a "partial degree of satisfaction" of some statement or condition (Zadeh 1965). Each rule represents fuzzy constraints at the boundaries between different rate classes for each input, such as cholesterol, blood pressure, or body-mass index.

Evolutionary Algorithms are also used in the new system, to optimize the numerical parameters in the fuzzy logic rules. The use of both Fuzzy Logic and Evolutionary Algorithms is described below.

As discussed above, Natural Language Processing techniques were also used to increase the capacity of the automated system.

Fuzzy Logic Rules Engine

The Fuzzy Logic Rules Engine was designed to handle discrete classification problems in which the decision categories form an ordered set (Bonissone et al. 2002). The FLRE was implemented within a Reusable, Optimizable Architecture for Decision Systems (ROADS), a generic framework designed at GE to implement intelligent decision engines (Aggour and Pavese 2003). The engine makes decisions through a 3-step process:

1. Rule evaluation through fuzzy membership functions
2. Aggregation evaluation and threshold application
3. Assignment of final decision (defuzzification)

A separate membership function is defined for each input for each rate class, to specify distinct cut-offs for each. Cut-offs were initially derived from knowledge engineering sessions with expert underwriters, and later optimized using an Evolutionary Algorithm.

When the FLRE makes a decision, the input data is passed through each of the fuzzy membership functions and scores are generated. After the rule scores have been generated, an aggregation is performed for each rate class. The scores are passed to each aggregation operation, which creates a single fuzzy score for each rate class in [0,1].

For each of these rate class scores, a pass/fail test is performed using a threshold value. Each rate class may specify different criteria for whether the tests pass or not. The rate classes are tested in the order of best to worst. The first rate class that passes all criteria becomes the final decision of the engine.

The FLRE is extremely flexible. Different membership functions can be defined for both continuous and discrete inputs. For continuous inputs, membership functions such as step (Boolean), trapezoidal, and Generalized Bell can be defined. For discrete inputs (such as binary), a fuzzy score can be associated with each value. Various functions can be used for the aggregation, including min, max, and average operations.

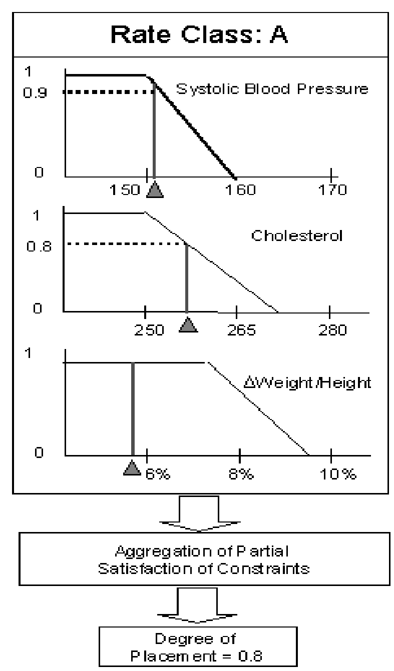


Figure 6: Fuzzy Rule Evaluation

Figure 6 shows a representation of three rules for one rate class, referred to as Rate Class A. For this example, the membership functions are trapezoidal and the aggregation is a min operation. The final step is for the engine to determine if the score of 0.8 falls within the threshold for Rate Class A. If it does, the applicant is assigned to this rate class.

Evolutionary Algorithm Optimization

The FLRE uses an Evolutionary Algorithm (EA) provided within ROADS for automated parameter tuning. Each chromosome in the EA contains a vector of tunable FLRE parameters. These elements typically represent membership function parameters (core and support values, for example), aggregation parameters, and threshold values. It is up to the system designer to specify what parameters to tune and what values remain static. Any subset of the parameters may be tuned at the discretion of the user—the ROADS EA generates the chromosome structure based on values set in an XML configuration file and loaded at runtime.

Since a chromosome defines a complete configuration of the FLRE, a new instance of the FLRE can be initialized and evaluated for each. At each generation of the EA, each chromosome in the current population initializes a separate instance of the FLRE. The engine is then run against a set of test cases. Each of these test cases must have a benchmark decision associated with it. The engine decision is compared to the benchmark decision, and a fitness measure of the engine's performance over the test set is calculated. This fitness function is used to rank the chromosomes in the population, determining how likely each is to be selected for crossover and mutation.

System Description

The automated underwriter system has a number of components, all executing in Microsoft Windows 2000 environments. As the summarizers digitize applications through their Web interface, the digitized information is stored in an Oracle database for further processing.

Every 15 minutes, a process is initiated that queries the database for any new cases. If the summarizers have entered a new case, it is extracted from the database, the appropriate FLRE is instantiated, and the case is processed through the appropriate engine. The output is then stored to the same Oracle database.

The FLRE was implemented entirely in Java 1.3.1 so that it can run in both UNIX and Microsoft environments without requiring re-coding. Once initialized, the engine takes fractions of a second to execute each case. The engine was designed and developed entirely in-house. Third-party tools were reviewed, but at the time none had the desired flexibility to represent underwriter knowledge in fuzzy rules that could be aggregated and tested against a threshold.

While multiple rules exist per rate class, repeated rule chaining was not allowed out of concerns for maintainability and readability. If a rule's result is an input to a second rule, then the output of the second rule cannot be used as input to any other rule.

Application Use And Payoff

Generation 1 was deployed in December 2002. It automated 12% of the LTC underwriting volume. Generation 2 was deployed in May 2004, increasing the percentage of automated applications to 19.2%. 100% of the applications are now digitized and sent to the APP-FLRE. In 2004, the average weekly volume sent to the APP-FLRE was 3,500 applications. Accuracy on the automated applications is near 100%. Generation 3 has been coded, is currently being tested, and is scheduled to go into production in 2005.

Before this system, 14% of all PHIs ordered were never used. The underwriters are now prevented from ordering PHIs and the engine orders only what is needed. Assuming the underwriters would have continued ordering at the same error level, the savings calculate to approximately $500K per year.

Automating this process had a number of other benefits, including improving decision consistency and significantly reducing the number of incorrect decisions. Reducing defects allows GNW to remain price competitive while effectively managing risk. And with an efficient, automated process handling a portion of the case volume, the capacity of the underwriting organization has increased.

In May of 2004, Genworth Financial was spun off from the General Electric Company. At the time of the IPO, stock analysts specifically cited this advanced technology as one of the key advantages GNW has over its competitors.

Application Development and Deployment

The FLRE was designed and developed by four engineers over a period of six months. The underwriter guidelines were collected initially from the underwriter manual, and then reviewed and updated with a committee of two underwriters and GNW's medical expert, requiring roughly three months of effort. The spiral development model was followed for the design and development of the FLRE and the implementation of the underwriter rules in the engine. The summarizer form creation and testing required about two months of effort from one engineer, two underwriters, and three representatives of the summarization team. The prototyping development model was followed for the implementation of the summarizer forms, as they required numerous iterations.

Data collection and validation took approximately four months for two of GNW's IT professionals. By far the most difficult step in the process was data collection and cleaning. Historical data was readily available to validate the decision engine and test the end-to-end process, but the quality of that data was less than ideal. Some cases were incomplete, and others did not have associated final decisions. A key takeaway for the team was to never underestimate the amount of time and effort required for handling data issues.

A significant amount of time and effort was invested by a diverse group to design, implement, and deploy the complete end-to-end system. While the use of AI technology was critical to the success of the project, it was only a component of the new system. Over $1 million was spent over the course of a year and a half to develop and test the end-to-end system. An additional four months was spent integrating the system into production.

The following process was followed for the development and deployment of each generation:

1. Knowledge acquisition from underwriter manual and review of guidelines
2. Transform guidelines into manually-tuned rules
3. Review rules with experts and users
4. Code rules and summarizer entry forms
5. Test on 100 examples
6. Review results with experts
7. Optimize rules with EA and update forms
8. Work with IT to install new rules and forms
9. Test on 400 more examples
10. Optimize rules with EA and update forms
11. Write training material
12. Release to pilot group
13. Review results of pilot
14. Update rules and forms
15. Finalize training material
16. Release to production
17. Sample 5% of volume processed
18. Monthly review of sample

This process ensures that (a) rules are never placed into production without a thorough evaluation, and (b) after release they are reviewed to ensure they are performing as expected.

Sixteen patents have been submitted to the U.S. Patent and Trademark office, covering many aspects of the automated underwriting process. These include:

1. System for Summarizing Information for Insurance Underwriting Suitable for Use by an Automated System
2. System for Rule-Based Insurance Underwriting Suitable for Use by an Automated System
3. System for Case-Based Insurance Underwriting Suitable for Use by an Automated System
4. System for Optimization of Insurance Underwriting Suitable for Use by an Automated System
5. System for Determining a Confidence Factor for Insurance Underwriting Suitable for Use by an Automated System

Maintenance

The system is maintained in three ways:

1. Major updates are made with every generation deployed
2. Minor updates are deployed between major updates
3. Parameter tuning can be performed with the evolutionary algorithm

Since the LTC underwriting rules do not change often, the majority of changes have been included with the generation releases.

If a change is made to the underwriting guidelines, the maintenance team can also deploy changes to the FLREs between generations. However, the primary reason for changing the underwriting guidelines has been clarifications needed to create rules from the guidelines in the first place. These clarifications in the guidelines are a side benefit of constructing the FLREs. Between-generation changes go through the thorough testing process described in the application development and deployment section.

Conclusions and Future Work

The automation of the underwriting of insurance applications has been a success. The artificial intelligence components (fuzzy logic rules engine, evolutionary algorithm, and natural language processing) enabled this success, but they were just one portion of the changes needed. This project required updating the underwriting guidelines, changing the underwriting process, switching the application process from paper-based to digital, adding personnel to digitize the summaries, and automating the creation of notification letters. The AI techniques were useful because they were a part of a larger end-to-end system.

In the future, FLREs for other impairments are planned in the order of the value of their addition, where the value is the cost of the current manual process minus the cost of creating, maintaining, and utilizing the forms and rule sets. Another group in GNW is creating a Web-based customer self-service application that will use the FLREs to give immediate rate quotes when all of the required data is available.

Acknowledgements

This work was a team effort with contributions from Piero Bonissone, Helena Goldfarb, Barry Hathaway, Dick Messmer and Marc Pavese at GE, and Amy Chambers, Dilip Chemburkar, Evely Euripides, David Gorman, Scott Hansen, JoAnn Hurley-Tuel, Bruce Margolis, Brian McCutcheon, Craig Robinson, Sylvia Soden, Rick Storms, Tammy Wood and Jerry Zeng at Genworth.

References

Aggour, K.S. and Pavese, M., 2003, ROADS: A Reusable, Optimizable Architecture for Decision Systems, *Proceedings of the 15th International Conference on Software Engineering and Knowledge Engineering*, pp 297-305

Bonissone, P.P., Subbu, R. and Aggour, K.S., 2002, Evolutionary Optimization of Fuzzy Decision Systems for Automated Insurance Underwriting, *Proceedings of the 2002 IEEE International Conference on Fuzzy Systems*, vol. 2, pp 1003-1008

Chisholm, M., 2004, How to Build a Business Rules Engine, Morgan Kaufmann

Jang, R., Sun, C., and Mizutani, E., 1997, Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence, Prentice Hall

Jurafsky, D., and Martin, J., 2000, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, Prentice Hall

McDonald, D.W., Pepe, C.O., Bowers, H.M., Dombroski, E.J., 1997, Desktop Underwriter: Fannie Mae's Automated Mortgage Underwriting Expert System, *Proceedings of the Ninth Conference on Innovative Applications of Artificial Intelligence*, pp 875-882

Talebzadeh, H., Mandutianu, S., Winner, C.F., 1994, Countrywide Loan Underwriting Expert System, *Proceedings of the Sixth Conference on Innovative Applications of Artificial Intelligence*, pp 141-152

Zadeh, L., 1965, Fuzzy Sets, *Information and Control*, vol. 8, pp 338-353