



Movie forecast Guru: A Web-based DSS for Hollywood managers

Dursun Delen^{*}, Ramesh Sharda, Prajeeb Kumar

Department of MSIS, Oklahoma State University, Stillwater, OK, United States

Available online 15 August 2005

Abstract

Herein we describe a Web-based DSS to help Hollywood managers make better decisions on important movie characteristics, such as, genre, super stars, technical effects, release time, etc. These parameters are used to build prediction models to classify a movie in one of nine success categories, from a “flop” to a “blockbuster”. The system employs a number of traditional and non-traditional prediction models as distributed independent experts, implemented as Web services. The paper describes the purpose and the architecture of the system, the development environment, the user assessment results, and the lessons learned as they relate to Web-based DSS development.

© 2005 Elsevier B.V. All rights reserved.

Keywords: Decision support systems; Web-based; Forecasting; Box-office receipts; Information fusion; Sensitivity analysis; Usability assessment

1. Introduction

In the motion picture industry, where the results of managerial decisions are measured in millions of dollars, managers are expected to make the best decisions in the shortest possible time. Success (or mere survivability) largely depends on quickly aligning the organizational resources towards changing market conditions in order to meet (and exceed) the actual (and perceived) needs and wants of the consumers. In order to succeed in such an unforgiving environment, managers (and other decision makers) need all the

help they can get. Decision support systems (DSS) can provide this much needed help. DSS (recently, also termed as business intelligence systems) are computer technology solutions that can be used to support complex decision-making and problem solving tasks [46]. A Web-based DSS, specifically, is a computerized system that delivers decision support information and decision support tools to managers (decision makers) via a “thin client” Web browser [38].

Prediction of financial success of a movie is arguably the most important piece of information needed by decision makers in the motion picture industry. Knowledge about the main factors affecting the financial success of a movie (and their level of influence) would be of great use in making investment and production related decisions. However, forecasting financial success (box-office receipts) of a particular

^{*} Corresponding author. Tel.: +1 918 594 8283; fax: +1 918 594 8281.

E-mail address: Delen@okstate.edu (D. Delen).

motion picture is considered to be a rather difficult and challenging problem. To some "... Hollywood is the land of hunches and the wild guesses" [27] due to the uncertainty associated with predicting the product demand. In support of such observations, Jack Valenti, current president and CEO of the Motion Picture Association of America, once mentioned that "... No one can tell you how a movie is going to do in the marketplace... not until the film opens in darkened theatre and sparks fly up between the screen and the audience" [51]. Trade journals and magazines of the motion picture industry have been full of examples, statements, and experiences that support such a claim.

The difficulty associated with the unpredictable nature of the problem domain has intrigued researchers to develop models for understanding and hopefully forecasting the financial success of motion pictures. Litman and Ahn [27] summarize and compare some of the major studies on predicting financial success of motion pictures. Most analysts have tried to predict the box-office receipts of motion pictures after a movie's initial theatrical release [26,44]. Because they attempt to determine how a movie is going to do based on the early financial figures, the results are not usable to make investment and have had production related decisions, which are to be made during the planning phase. Some studies have attempted to forecast the performance of a movie before it is released but had only limited success. These previous studies, which are either good for predicting the financial success of a movie after its initial theatrical release or are not accurate enough predictors for decision support, leave us with an unsatisfied need for a forecasting system capable of making a prediction prior to a movie's theatrical release. Our research aims to fill this need by developing and embedding an information fusion-based forecasting engine into a Web-based decision support system that could be used by Hollywood managers.

The paper is organized as follows. Section 1.1 gives an overview of Movie Forecast Guru. Section 2 gives a brief review of the related literature in forecasting financial success of motion pictures and Web-based DSS. Section 3 presents our conceptual architecture along with specifics about its implementation and a brief description of the major

modeling components. Section 4 illustrates the use of the system by providing a walk-through using an interaction flow diagram. The method used and the results obtained regarding the user evaluation of MFG are also presented in this section. The paper concludes with Section 5 where a summary of the findings along with the limitations and future directions of the research are identified. Appendix A includes detailed screen shots of interaction with MFG.

1.1. MFG: Web-based DSS overview

Fig. 1 illustrates the conceptual architecture of the Movie Forecast Guru (MFG in short) at a very high level. MFG is a Web-based DSS capable of responding to user requests initiated from within a Web browser. Its engine resides in a Web server and is capable of using data (local and remote), models (local and remote) and a knowledge base to carry out its duties: generating financial success predictions and providing sensitivity analysis on the parameters for a variety of movie scenarios generated by the decision makers (investors, producers, distributors and exhibitors). Each of the prediction models used is implemented as a separate Web service, representing an expert available on demand. The core engine can consult each expert, and can present the results of the individual experts as well as a combined forecast to the user. Compiled data from previous performance can also be fed back to the individual models to improve their forecasting performance. The scenarios evaluated by a user are stored in a database for further analysis and/or reuse.

MFG is implemented as a Web-based DSS as opposed to a "desktop application" for a number of reasons:

- **Distributed computing**—Web-technology enables us to develop the system in such a way that it has a single point of access/entry (front-end), yet provides the means to access a large number of external links (models and data sources) to construct the content in the back-end. Therefore, the complications of the information creation process is hidden from the end user (decision maker) by encapsulating the details within the web server engine and providing the end user only the information they

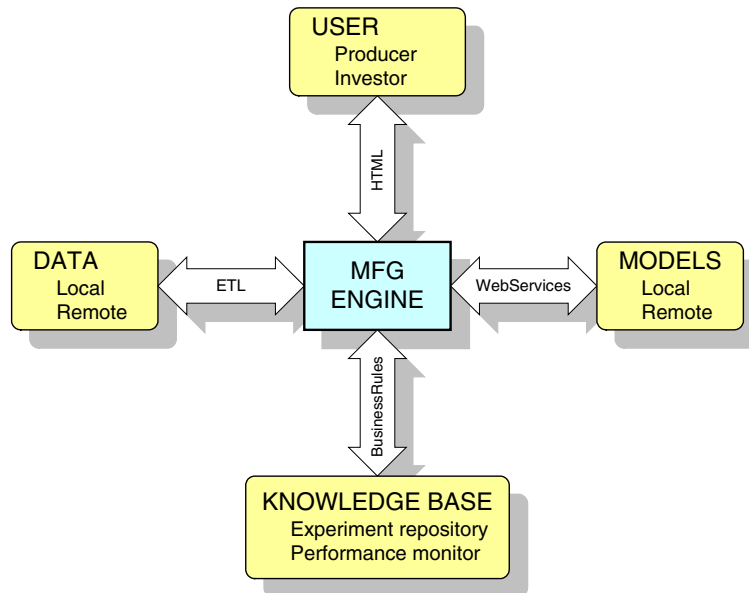


Fig. 1. A high-level conceptual architecture for MFG.

need to make decisions in an understandable multimedia (graphical and dynamic) format.

- **Versioning**—With the help of a Web-based infrastructure, the end user is likely to have access to the latest version of the MFG system. In contrast, keeping a client application (a desktop application) up to date with the current version would be quite burdensome, since the models are continuously updated as new data and models become available.
- **Platform independence**—Web-based DSS can be developed independent of the type and nature of the client's computing environment. This allows the development team to spend more time on advancing the underlying methodology of the system as opposed to translating the client application into different versions so that it can run on a wide variety of possible computing platforms. It is especially true in this application where the diversity of computing platforms is evident. The business side of studios may use Windows-based computers, whereas the artistic community may use Macintosh or other graphics intensive platforms.
- **Use of models not owned or developed by the system owner**—Sophisticated prediction models might be maintained at distant/proprietary loca-

tions. The owner of the system might not own the models but have access privileges to use them via some type of a subscription system. With the advent of the Web and its enabling technologies such as the Web services, this kind of computing infrastructure is becoming more and more popular. These external models can also be thought of as human experts. In fact, in the future we plan to add human expert along with the sophisticated analytical models in our “expert” arsenal so that we can provide the decision makes with the most accurate forecast. This class of distributed, integrated infrastructure utilizing multi-expert prediction system is very hard (if not impossible) to implement using traditional desktop applications.

- **Facilitating collaboration among stakeholders**—The Web-based DSS approach is also capable of supporting multiple decision makers (i.e., stakeholders, namely, investors, producers, distributors, and exhibitors) allowing them to interact with each other using the MFG forecasting engine from distant locations. Such infrastructure provides a desirable platform for group decision-making in arriving at a consensus where each stakeholder could adjust the parameters of their preferred fore-

casting model of a potential movie project until a consensus is reached or some remedial action is identified among the stakeholders with respect to planning of the movie.

2. Background

This section aims to present a brief review of the relevant literature. First, research related to predicting the financial success of motion pictures is cited. Then, recently published research in Web-based DSS is summarized.

2.1. Background on forecasting financial success of Hollywood movies

Not much has been reported in the literature with respect to research in decision support systems in motion picture industry. So far, most research efforts have been limited to developing a single model to predict the box-office receipts and/or to explain the decision variables. The literature on forecasting financial success of new motion pictures can be classified based on the timing of the forecast: (i) Before the Initial Release—that is, forecasting the financial success of the movies before their initial theatrical release [11,15,26,28,49,56], and (ii) After the Initial Release—that is forecasting the financial success of the movies after their initial theatrical release, when the first week of box-office receipts are known [32,42,44]. Forecasting models that fall into the category of “after the initial release” tend to generate more accurate forecasting results due to the fact that those models have more explanatory information including box-office receipts from the first week of viewership, movie critic reviews, and word-of-mouth effects. Our own work in exploring the use of neural networks for predicting the box office performance before a movie is released (or indeed, produced) has shown promise [34]. However, these models are not available to the end-users in an implementable form. The present project aims to bring several of these models to the user in an implementable system.

From the standpoint of a DSS type of implementation in the motion picture industry, the only notable study comes from Eliashberg et al. [16]. They developed a marketing management support system

(called SilverScreener) that helps theatre managers to choose among alternative movies to show in their theatres. Their system is based on a multi-phase mathematical attendance-forecasting model. The model was developed using the historical data from Pathé theatres in Netherlands, along with managerial judgment and theatre-specific factors. Their limited tests on the same theatre indicated that the overall financial success (total revenues) of the theatre increased compared to two other theatres in the same time period. They also reported improved managerial attitudes towards the system once the system started to generate better decisions resulting in higher revenues. However, in contrast to MFG, this system was not meant for decision support in pre-production stage.

Given the significant investment in movie industry, the need for substantial decision support is evident. A general lack of published research in DSS applications in this domain suggests opportunities for much research. This paper is a step in this direction.

2.2. Background on Web-based DSS

The Web has become a ubiquitous communication medium [35]. Much has been accomplished in using the Web for information publishing, conducting and supporting online shopping, and designing virtual organizations that support online shopping activities [50]. In addition to e-Commerce and related uses, the Web has also been used as a mechanism to deliver information and knowledge to managers and other knowledge workers. For instance, recent literature has described Web-based systems to support decision makers in the healthcare industry [7,23]; manufacturing and supply-chain management [2,22,24,25]; financial portfolio selection [5,12]; law enforcement [30,54,55]; and defense [52].

According to Power and Kaparathi [39], the World Wide Web is where the action is in developing decision support systems. Web-based DSS have reduced technological barriers and made it easier and less costly to serve decision-relevant timely information (via decision support tools) to managers wherever and whenever they may need it. A well-designed Web-based DSS should increase the use of DSS tools and techniques in making critical managerial decisions [38], largely because it has the poten-

tial to make access to organizational knowledge repositories fast, easy and inexpensive. Because the Web can reduce some of the technical problems associated with the hardware and software issues (e.g., need for special hardware configuration and software components to install/run enterprise-wide applications on the client machines), it is considered an excellent platform for deploying the next generation DSS applications.

Exemplars of Web-based decision support system implementations are included at the DSSResources.com site [14] (a repository of a number of model-driven, Web-based applications). In addition, several other Web-enabled systems have been implemented to facilitate interaction of models and/or data through the web. These include Open Optimization Framework—OOF [36], the NEOS project [33], the Optimization Service Provider—OSP [37], WEBOPT [53] (a list of Web-enabled optimization modeling

applications) and AURORA [3] (a Web-enabled high performance computing site for financial applications). These systems illustrate Web-enabling of the dialog, data, and model sub-systems of decision support systems at various levels. To the best of our knowledge, there is no reported study that claims to have developed Web-based DSS for Hollywood managers. Thus our system fills a specific niche for this industry. Another emerging trend in the Web-based DSS is the distributed modeling (and fusion of multi-model results) lets the decision maker see the results obtained from multiple models not necessarily located in the same computing environment. Though it is possible to build such multi-model infrastructure in a “classic” desktop application, the Web offers a greater likelihood of utilizing the latest models (both local and remote ones) in real time application mode. Section 3.4 describes this further.

3. Implementation of the movie forecast Guru

This section provides details about the implementation of the MFG application. Section 3.1 specifies the problem, the terms of the data and the variables used to develop the models are given. Section 3.2 lists and briefly describes the prediction models used, Section 3.3 presents the software architecture and explains the communications/interactions among the software components, and Section 3.4 discusses the use of Web services as an implementation framework for the distributed model management component of MFG.

3.1. Problem description

The dependent variable in our study is the box-office gross revenues. We classify a movie based on its box-office receipts in one of nine categories, ranging from a “flop” to a “blockbuster.” This process of converting a continuous variable in a limited number of classes is commonly called as “discretization”. Many prefer using discrete values as opposed to continuous ones in prediction modeling because (i) discrete values are closer to a knowledge-level representation [48]; (ii) data can be reduced and simplified through discretization [29], (iii) for both users and experts, discrete features are easier to understand, use, and explain [29]; and finally, (iv) discretization makes many learning algorithms faster and more accurate [13]. We discretized the dependent variable into nine classes using the following breakpoints:

Class no.	1	2	3	4	5	6	7	8	9
Range	<1	>1	>10	>20	>40	>65	>100	>150	>200
(in Millions)	(Flop)	<10	<20	<40	<65	<100	<150	<200	(Blockbuster)

We used seven different types of independent variables. Our choice of independent variables is based on previous studies conducted in this domain. The list of variables, their definitions along with their possible values is listed in Table 1.

Table 1
Summary of independent variables

Independent variable name	Definition	No. of possible values	Range of possible values
MPAA Rating	The rating assigned by the Motion Picture Association of America (MPAA).	5	G, PG, PG-13, R, NR
Competition	Indicates the level at which each movie competes for the same pool of entertainment dollars against movies released at the same time.	3	High, Medium, Low
Star value	Signifies the presence of any box office superstars in the cast. A superstar actor/actress can be defined as one who contributes significantly to the up-front sale of the movie.	3	High, Medium, Low
Genre	Specifies the content category the movie belongs to. Unlike the other categorical variables, a movie can be classified in more than one content category at the same time (e.g., action as well as comedy). Therefore, each content category is represented with a separate binary variable.	$36 \frac{(9!/7! 2!)}{72/2} = 36$	Sci-Fi, Epic Drama, Modern Drama, Thriller, Horror, Comedy, Cartoon, Action, Documentary
Special effects	Signifies the level of technical content and special effects (animations, sound, visual effects) used in the movie.	3	High, Medium, Low
Sequel	Specifies whether a movie is a sequel (value of 1) or not (value of 0).	2	Yes, No
Number of screens	Indicates the number of screens on which the movie is planned to be shown during its initial launch.	3876	A positive integer between 1 and 3876

3.2. Prediction models and the dataset

We built four different types of models: Neural Networks, Decision Trees, Ordinal Logistic Regression and Discriminant analysis. These models were selected for inclusions in this DSS due to their acceptable performance in our earlier comparisons [34]. We also used an information fusion meta-model that generates predictions using the output of all four individual models. Our models were estimated using a dataset of 849 movies released between 1998 and 2002. The data was drawn from ShowBiz Inc. [47], Internet Movie Database (<http://www.imdb.com>) and <http://www.the-numbers.com>, and synchronized based on movie names. What follows is a short description of these model types and their specific implementations for this research.

3.2.1. Neural networks

Neural networks are commonly known as biologically inspired, highly sophisticated analytical techniques, capable of modeling extremely complex non-linear functions. Formally defined, neural networks are analytic techniques modeled after the processes of learning in the cognitive system and the neurological functions of the brain and capable of predicting new observations (on specific variables) from other observations (on the same or other variables) after executing a process of so-called learning from existing data [20]. We used a popular neural network architecture called Multi-Layer Perceptron (MLP) with back-propagation algorithm. MLP is essentially the collection of nonlinear neurons (perceptrons) organized and connected to each other in a feedforward multi-layer structure. MLP is known to be a strong function approximator for prediction and classification problems. It is arguably the most commonly used and well-studied NN architecture. Hornik et al. [21] empirically show that given the right size and the structure, MLP is capable of learning arbitrarily

complex nonlinear functions to an arbitrary accuracy level. Fig. 2 shows the graphical representation of the MLP used in this study.

3.2.2. Decision trees

Decision trees are powerful classification algorithms that are becoming increasingly popular with the growth of data mining in the IS field. Popular decision tree algorithms include Quinlan’s ID3, C4.5, and C5 [40,41], and Breiman’s CART [6]. As the name implies, this technique recursively separates observations in branches to construct a tree for the purpose of improving the prediction accuracy. In doing so, different mathematical algorithms (e.g., information gain, Gini index, and Chi-squared test) are used to identify a variable and the corresponding threshold for the variable that splits the pool of observations into two or more subgroups. This step is repeated at each leaf node until the complete tree is constructed. The objective of the splitting algorithm is to find a variable-threshold pair that maximizes the homogeneity (order) of the resulting two or more subgroups of samples. The most commonly used mathematical algorithms for splitting include Entropy based information gain (used in ID3, C4.5, C5), Gini index (used in CART), and Chi-Squared test.

3.2.3. Logistic regression

Logistic regression is a generalization of linear regression [19]. It is used primarily for predicting binary or multi-class dependent variables. Because the response variable is discrete, it cannot be modeled directly by linear regression. Therefore, rather than predicting point estimate of the event itself, it builds a model to predict the odds of its occurrence. In a two-class problem, odds greater than 50% would mean that the case is assigned to the class designated as “1” and “0” otherwise. While logistic regression is a very powerful modeling tool, it assumes that the response variable is linear in the coefficients of the predictor variables. Furthermore, the modeler, based on his or her experience with the data and data analysis, must choose the right inputs and specify their functional relationship to the response variable.

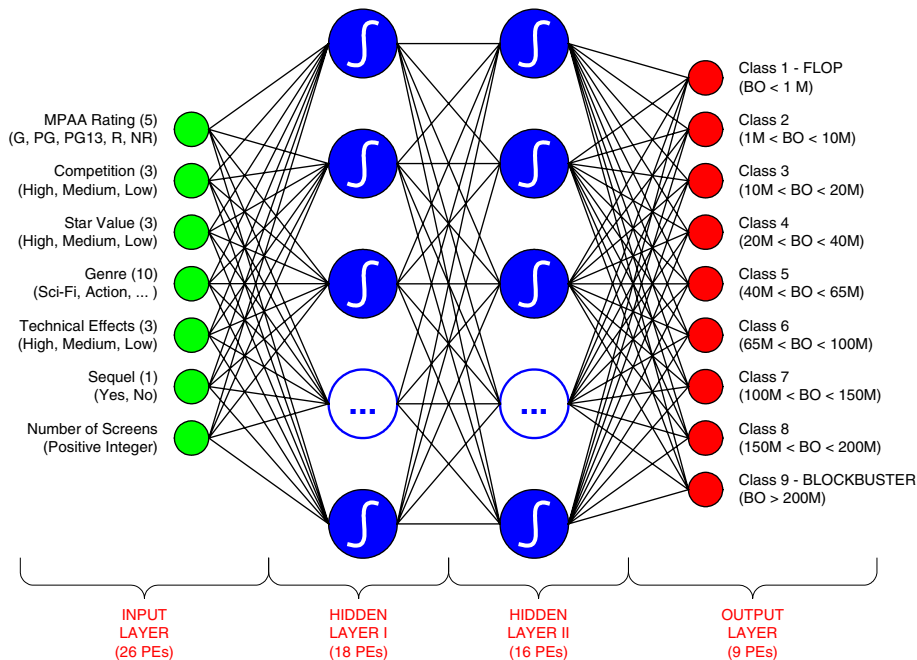


Fig. 2. Graphical representation of our MLP neural network model.

3.2.4. Discriminant analysis

Discriminant analysis is one of the oldest statistical classification techniques, first introduced by Fisher [17]. Using the historic data, it finds hyper planes (e.g., lines in two dimensions, planes in three, etc.) that separate the classes from each other. The resultant model is very easy to interpret because all the user has to do is determine on which side of the line (or hyper-plane) a point falls. Training is simple and scalable. Despite its scalability and simplicity, discriminant analysis is not a popular technique in data mining for two main reasons: (1) it assumes that all of the predictor variables are normally distributed (i.e., their histograms look like bell-shaped curves), which may not be the case, and (2) the boundaries that separate the classes are all linear forms (such as lines or planes), but sometimes the data just can't be separated that way.

3.2.5. Information fusion

Information fusion is the process of intelligently combining the information (forecasts/predictions in this case) created and provided by two or more information sources (forecasting models). While there is an ongoing debate about the sophistication level of the fusion methods to be employed, there is a general consensus that fusion (combining forecasts and/or predictions) produces more useful information for business decisions [1]. Combining forecasts can improve accuracy, completeness, and robustness of information, while reducing uncertainty and bias associated with individual modelers [8].

In prediction modeling, there is not a universally accepted “best model” that works for any problem. The best model depends on the scenario being analyzed and the data set being used; and can only be obtained through trial-and-error experimentation [43]. Just as there is not a single best model, there is also not a single best implementation of different model types. Researchers are developing new ways to improve the accuracy and efficiency of prediction models. Therefore, it would be desirable to combine the models developed by multiple experts [4]. Use of multiple experts should make the forecasts (both individual and combined) more accurate and more efficient. MFG has the potential of taking full advantage of using multiple experts and the models developed by them. This multi-model fusion algorithm can be mathematically illustrated as follows:

Given the expected response variable (y) and the decision variables (x_1, x_2, \dots, x_n) the formulation for any prediction model can be written as Eq. (1)

$$\hat{y} = f(x_1, x_2, \dots, x_n). \quad (1)$$

Prediction model f can take many forms. For instance, a linear regression model can be written as Eq. (2)

$$f(x_1, x_2, \dots, x_n) = \beta + \sum_{i=1}^n a_i x_i \quad (2)$$

where β is the intercept and a_i 's are the coefficients for x_i 's. For a Neural Network model, for a single neuron, it may be written as Eq. (3)

$$f(x_1, x_2, \dots, x_n) = \phi \left(w_0 + \sum_{j=1}^n w_j x_j \right) \quad (3)$$

where ϕ is the transfer function and w_i 's are the weights for x_i 's. Given that we use m number of prediction models, the fusion model can be written as

$$\hat{y}_{\text{fused}} = \psi(\hat{y}_{\text{individual},i}) = \psi(f_1(x), f_2(x), \dots, f_m(x)) \quad (4)$$

If ψ is a linear function, which is the case in this study, then we can write Eq. (4) as

$$\hat{y}_{\text{fused}} = \sum_{i=1}^m \omega_i f_i(x) = \omega_1 f_1(x) + \omega_2 f_2(x) + \dots + \omega_m f_m(x) \quad (5)$$

where

$$\sum_{i=1}^n \omega_i = 1. \tag{6}$$

The values for ω s are derived from the up-to-now prediction accuracy measure of the individual predictors. That is, the higher the accuracy of a predictor on independent test cases gets, the larger the weight that is assigned to that predictor type.

MFG implements all of the above methods as independent expert models. The models have initially been trained using data from the last five years. As mentioned earlier, each model is then run in prediction mode as a Web service.

3.3. Software architecture

Software architecture for MFG is presented in Fig. 3. As is the case for most web applications, MFG is built on a three-tier architecture. It follows and evolves from the Web-based DSS architecture presented in Power and Kaparathi [39]. In the first tier lies the user interface in the form of a “thin client” Web browser (typically Microsoft Internet Explorer or Netscape Navigator). From the standpoint of client software requirements, MFG does not require the client to install any additional software components to be able to use the MFG system. All necessary software components are server based. In the second tier, MFG has the core of the architecture called MFG Engine. MFG engine is a collection of software procedures written in ASP.NET and hosted on a Microsoft IIS Web server. The communication between the user interface and the engine is accomplished through HTTP.

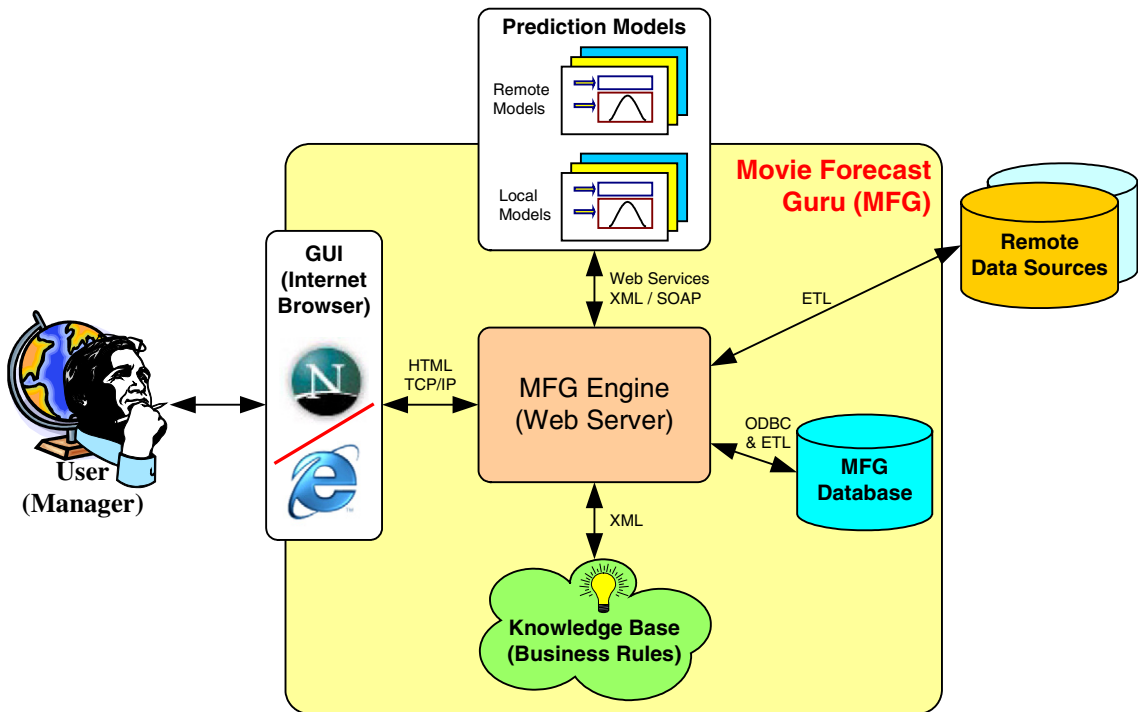


Fig. 3. Software architecture of MFG.

The third tier is the MFG database where the user, movie, and experiment related data are stored. MFG engine has the capability to access external data set sources to periodically collect domain-related data using extract-transport-load (ETL) procedures. This external data becomes a part of the movie database and is used to (i) test the accuracy of the individual prediction models and the system itself, and (ii) develop new models and/or refine existing prediction models. MFG engine also uses models that are outside the Web server it is built on. That is, it has the capability to access prediction models that reside in remote servers. MFG engine uses Web services along with XML and SOAP to carry out this task. Knowledge base, a collection of business rules, plays the role of managing and governing the interactions between the individual tiers as well as the interactions with the remote models and data sources. The knowledge base also has the business rules that continuously monitor the accuracy and currency of the individual prediction models, and recreate or refine them on an as needed basis. The following subsection provides a summary of MFG implementation.

3.4. Use of web services as model component of MFG

A commonly referenced software architecture for Web-based DSS is presented in Power and Kaparathi [39]. MFG uses this architecture as a starting point and evolves from it in two directions: (1) it uses a distributed DSS framework where each prediction model is implemented as a separate Web service (which enables the system to be more flexible and expandable towards being receptive to new prediction models), and (2) it incorporates capabilities of information fusion to combine the results of multiple prediction models (which would allow the system to produce more accurate and less biased estimates). These architectural enhancements should be viewed as stepping-stones towards achieving the nirvana of truly expandable, highly adaptive, and distributed (yet seamlessly integrated) Web-based DSS of the future.

Within the architecture of MFG, the model management component is designed to be flexible (new model types can be added to the system without changing the code of other parts of the system), modular (all models are developed as stand alone components so that they can be unplugged and re-plugged as needed) and adaptable (prediction capabilities of models are monitored over time to detect deterioration and maintained accordingly). In order to accomplish these goals, every prediction model in MFG is designed and implemented as a standalone software component. Use of component-based model infrastructure for Web-based DSS is not new. Gunter et al. [18] proposed a model management system for sharing statistical computing modules. They developed a middleware that facilitates the communication between remote applications and their computing modules. According to them, these modules could potentially be used as part of a large community of Web-based systems on remote locations. Since then, much has been accomplished in developing technologies (e.g., CORBA, XML, COM, DCOM, JavaBeans, and recently Web services) to make it easier between the two parties (applications on the demand side and software component providers on the supply side) to communicate with each other.

We used “Web services” to facilitate the distributed component-based model infrastructure. Although there are a number of varied and often seemingly inconsistent motivations for, and uses of, the term “Web service”, the following definition, which comes from W3C (<http://www.w3.org>), is the most commonly accepted one: “A Web service is a software system identified by a Uniform Resource Identifiers (URI), whose public interfaces and bindings are defined and described using XML, such that its API can automatically be discovered by other software systems.” These systems access Web services via ubiquitous Web protocols and data formats such as HTTP (Hypertext Transfer Protocol), XML (eXtensible Markup Language), and SOAP (Simple Object Access Protocol), without worrying about how each Web service is internally implemented. Web services can be accessed by a software application written in any language, using any component model, running on any operating system. XML, a markup language for documents containing structured information, is commonly used to format the input and output parameters of the request, so that the request is not tied to any particular component technology or object calling convention. MFG uses XML and SOAP to communicate with the models located in near and remote locations. These prediction models are developed using Visual Basic .NET and C# programming languages. C# is a new programming language introduced by Microsoft to bridge the gap that it felt existed in previous languages

[31]. C# is a component-oriented language that is designed to help developers do more with fewer lines of code and with fewer possibilities for errors. C# is built on the two widely used programming languages, C and C++. It derives its syntax, many of its keywords and operators from the C language, while is built upon the object model defined by C++ [45].

In MFG, a database driven central registry is used to keep track of the available prediction models and their characteristics. As new models are added to the system, the registry is updated. Each model is identified with its characteristics such as location (which server it resides in the form of an IP address), name (under what name it can be accessed in the form of a URI), security specification (user name and password required to access the component), and API (input/output requirements).

The prototype system of MFG has been implemented on Windows XP using Microsoft .NET environment. The database in this version of MFG is small enough to run under Microsoft Access. In the following section, we present user interaction and the usability assessment of the prototype implementation of MFG.

4. Usability assessment of MFG

Since MFG is built as a Web-based DSS, it is accessed by the end users through a Web browser. The primary design criteria for MFG were to make it intuitive and user friendly to its potential audience: Hollywood managers. The use of the MFG system is illustrated in Fig. 4 using a user interaction flow diagram. Additional details and screen shots are included in Appendix A.

4.1. Usability assessment

We conducted an exploratory assessment of the DSS. Fig. 5 shows the usability assessment model employed for this study. This model is consolidated from DeLone and McLean [10] and Davis [9] studies. DeLone and McLean's [10] IS success model is a manifestation of a large body of literature focused on the success of an information system implementation. Their model suggests that system quality and information quality affect the use and the user satisfaction. The use and user satisfaction affect each other and in combination affect the impact on an individual and eventually the organization. Davis [9] derived a 7-point scale to measure the acceptance of information technology in terms of perceived usefulness and perceived ease of use. We used these two studies in conducting our usability assessment of MFG. We built a ten-question survey instrument using a seven-point Likert scale to collect the evaluation of MFG system assessment from the end users.

The survey was operationalized using a Web-based user interface. A group of 187 students (a

mix of senior level undergraduate and first or second year graduate students) taking (or having recently taken) courses related to analytics-driven decision making such as advanced information systems development, advanced business systems, production operations management, advanced decision analysis, and decision support systems from the Management Science and Information Systems department at Oklahoma State University participated in the survey. The participants were given a half-hour presentation about the system, and then asked to use the system, run experiments, conduct sensitivity analyses, observe and evaluate the results, and then fill out the survey form. The form was located in the same Web environment as the MFG system for convenience purposes. Without leaving the MFG Web page, a user can fill out the form, submit the survey results, and let MFG automatically record them into the database.

The results of the user assessment are summarized in Fig. 6. The small square box shows the mean and the two vertical lines on both sides of the square box shows the $\pm 1\sigma$ (standard deviation) range. As the results indicate, the users rated the prototype implementation of MFG slightly-to-quite high on system quality, quite high on information quality, and quite high on usefulness. Based on the usability assessment model presented in Fig. 5, these results could lead to positive individual and organizational impact (which are not measured explicitly in this study).

Even though the results presented in Fig. 6 are somewhat self-explanatory, we note some interesting insights. A small standard deviation and one of the

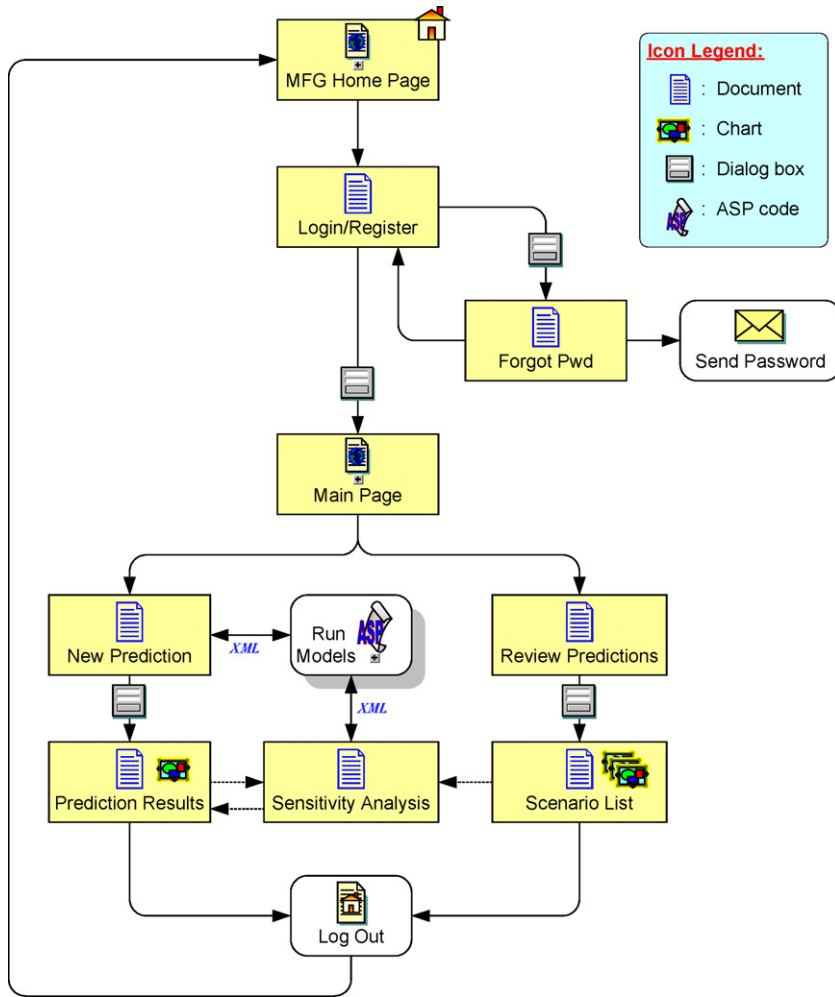


Fig. 4. User interaction flow diagram for MFG.

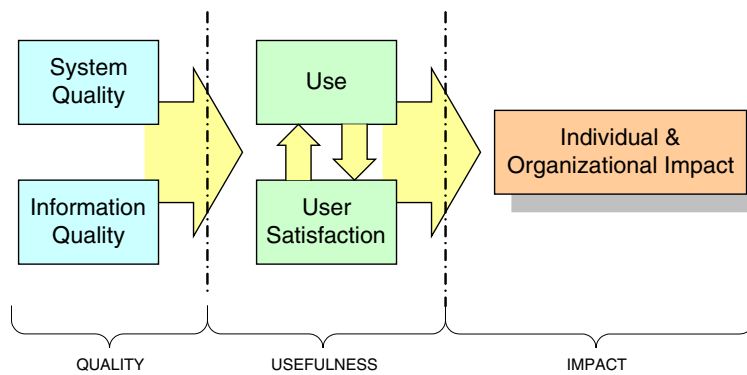


Fig. 5. Usability assessment model (adapted from DeLone and McLean [10]).

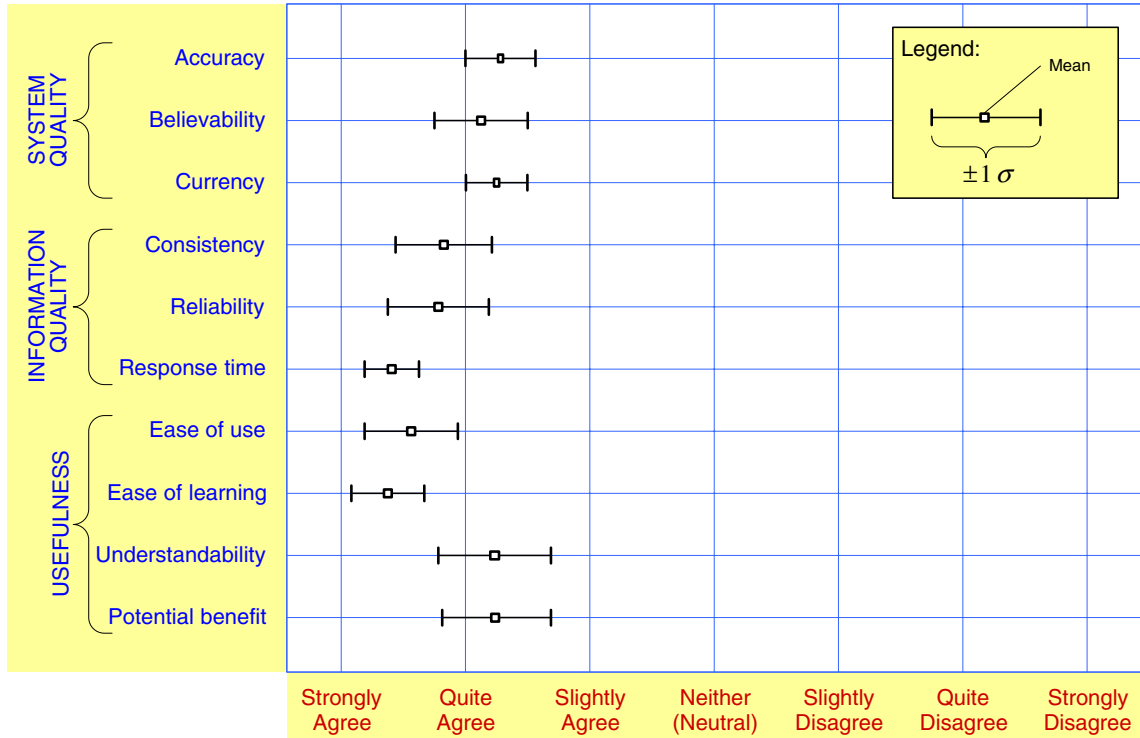


Fig. 6. Summary of the user assessment results.

best ratings for “Response time” indicate that use of Web services (and hence the distributed-modeling) may have improved the performance of the system. Web services can run on different servers and thus can be used in implementing a distributed Web-based DSS. Somewhat lower rating for “Understandability” criterion can be explained with “first time exposure syndrome.” Providing more help and further on-the-spot explanations along with having users use the system on a regular basis could improve this measure. Relatively lower rating for “Potential benefit” criterion may be explained by the lack of expertise in such decision making by our subjects.

Overall, this exploratory test of the system shows promise for this DSS application. We acknowledge the fact that using the college students as the subjects of this assessment limits the value of these results. However, it is common in the MIS/DSS literature to use students to test the basic functionality of a system. Though the problem domain of our DSS is highly specialized, it is also general and popular enough in that everyone can relate to the basic problem, and the

dependent and independent variables. In that sense, there is some validity of these results.

A more convincing and useful survey of the intended actual end users, the Hollywood managers, needs to be conducted. So far, our limited number of Hollywood contacts have provided us with encouraging reviews of our work and invaluable comments to help us design MFG to be a Web-based decision support tool that addresses the needs and wants of these users. Efforts are under way to organize formal tests by actual decision-makers at a major studio.

5. Summary and conclusions

The World Wide Web and the associated technologies have been evolving at an astonishing rate, creating an unprecedented opportunity for the DSS community to develop systems that can assist the decision makers in a cost effective manner. The combination of Internet technologies and DSS tools are manifesting them-

selves into Web-based information repositories where the organizational data, information and knowledge are stored and, on an as needed basis, delivered to the decision makers wherever and whenever they may need it. MFG, the Web-based DSS for Hollywood managers, is designed and developed to take full advantage of the latest technologies in Internet and in DSS. It uses a distributed modeling and model evaluation concept where each prediction of a model type represents an expert opinion. Each model is implemented as a Web service to make the MFG architecture flexible, expandable and adaptable. An information fusion model (also implemented as a Web service) is used to combine the results of all prediction models so that the estimates are more robust, more accurate and less biased. The framework has the potential to allow individual models (including

the ones for information fusion) to be adaptive in the sense that the models respond to the changes in the application domain by changing the parameters over time. To the best of our knowledge, this is the first Web-based decision support system developed to address the information needs of managers in motion picture industry.

Some of the generalizable characteristics of MFG's Web-based architecture are as follows: (1) Architectural flexibility: A contemporary Web-based DSS should adjust itself to changing characteristics (e.g., data, heuristics, decision models) of the underlying domain in order to provide the user with the latest and "the greatest" information every time it is used. This is accomplished in MFG by using Web-services as the representation of the distributed prediction experts, which allows MFG to easily approach them

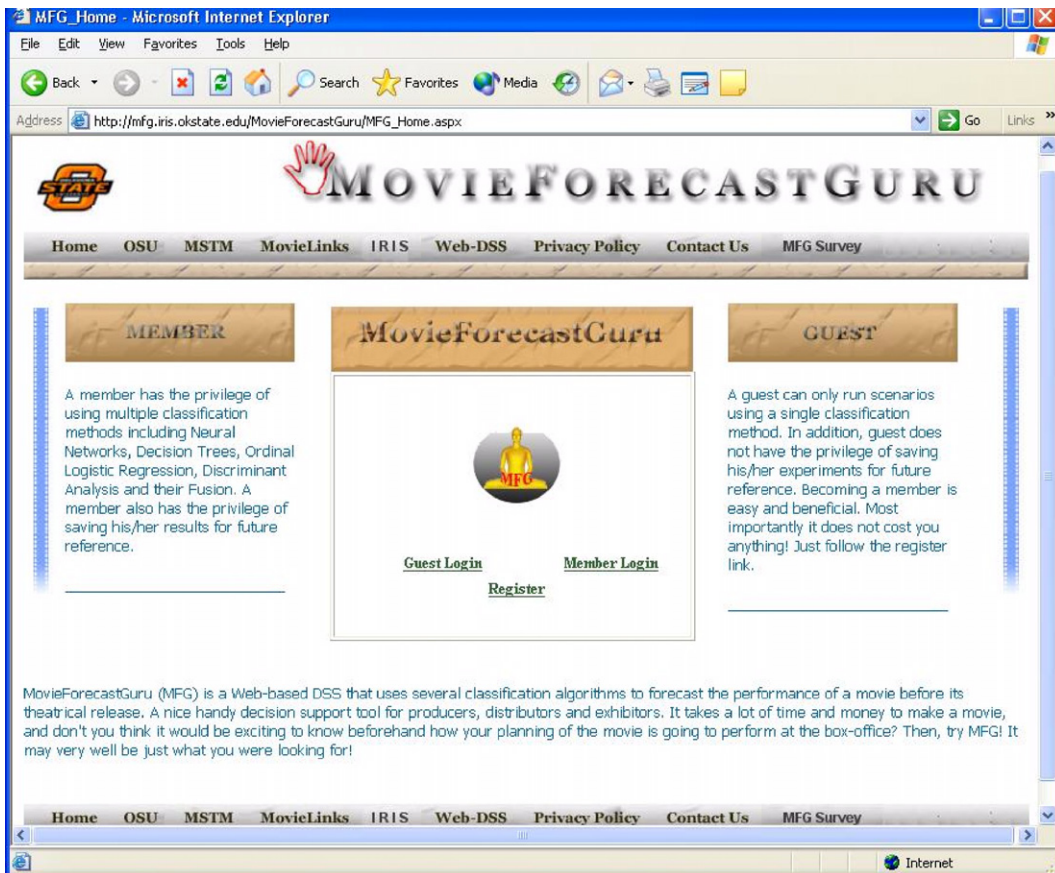


Fig. A1. MFG homepage.

wherever they may be located at the time of service requisition. (2) Expandable model set: A Web-based DSS that deals with the domains as unstructured as the one addressed in this paper should have the capability of adding new experts (and models) to its prediction arsenal as they become necessary/available without requiring user to rewrite the source code of the server site. This is accomplished with a generalized design of the relational database, where the models are managed. This design allows us to manage (add and delete) new Web-services (i.e., experts and analysis models) at the database level (insertion of a new record to the list of prediction experts table) as opposed to changing and re-compiling the entire DSS code.

There are several directions in which this work can be extended. The system can be enhanced to poll human expert opinions to improve the prediction models. For instance, if a significant number of users indicate that a model can be improved by including an additional parameter to capture a specific characteristic of the problem (e.g., level of correlation

between the movie script and the current social/political affairs), then the system should be receptive to such a proposal.

Once in use by the decision makers, the forecasted results obtained over time can be stored and matched (synchronized/compared) with the actual box-office data (as soon as they become available) to check how good the forecasts (both individual model predictions and the combined ones) had been. Based on the results, the new data can be used to update the parameters of the predictions models, moving towards realizing the concept of living models.

This system can potentially be used in making strategic as well as tactical decisions. For instance, a movie studio can use the prediction capabilities of MFG to make a “go no-go” decision on a proposed movie project (a strategic decision), or a producer can make a decision to go with a super-star (actor/actress) or go with an ordinary actor/actress (a tactical decision). Both of these decisions would use the information that can be obtained from



Fig. A2. MFG forecast parameter entry/selection interface.

sensitivity analysis. These “star value” decisions are not as trivial as one might think. For instance, common knowledge would suggest that as the star value of the cast increases, the financial success of the movie should also increase. This may be true most of the time, but there were instances where the star value did not add anything to the financial success of a movie. In fact, in some instances it actually hurt it. The relationships among the decision variables are quite complex and therefore, should be analyzed using a collection of sophisticated prediction models with a sensitivity analysis engine such as MFG.

MFG can be enhanced to serve as a group decision support system. Investing in a movie, choosing among alternatives parameter combinations for production project, and adopting a movie for viewership are all very complex decisions to make. They require careful considerations of many factors that are naturally uncertain. In fact, these kinds of unstructured decision are usually made by a group of people (as opposed to a single individual). MFG can potentially be used by a group of stakeholders (wherever they

may be at the time of collaboration) to interactively create a number of scenarios (what-if analyses) for a potential project before finalizing the production plans.

As these further research directions are integrated into the current implementation of MFG, the system would be more truly expandable, highly adaptive, fully automated, distributed Web-based DSS that is capable of providing information to and polling data from a community of domain users. In a sense, once realized, this system might be considered as a new type of DSS that could be termed a Community DSS (or CDSS in short). The CDSS would be used by a community of interest to receive support for their own decision-making and/or to provide input into refinement of such DSS.

Acknowledgements

We are thankful to Rahul Satyanarayan for his assistance in implementation of MFG, and to the

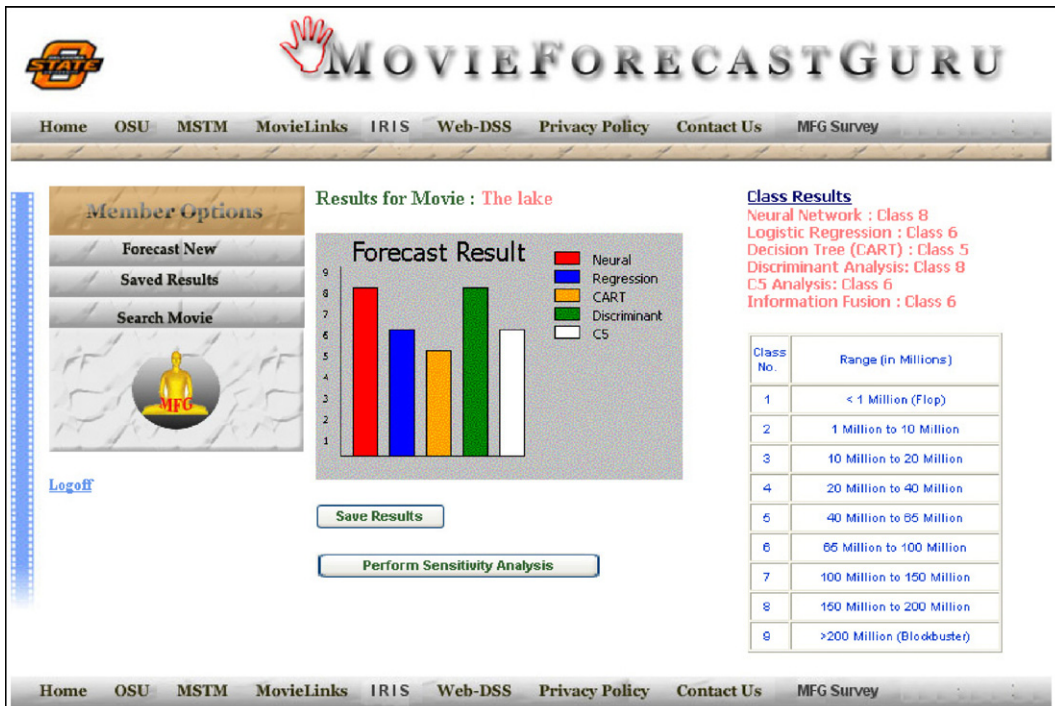


Fig. A3. MFG forecast result interface.

anonymous referees and the associate editor for helpful comments on earlier drafts of the paper.

Appendix A. User interaction

The MFG system is accessed through a Web browser. Once in the MFG homepage (Fig. A1), the user is presented with the options of (1) log in as a user who previously registered for the site, (2) register to become a regular user, and (3) log in as a guest. The advantages of becoming a registered user are (1) registered users are able to save the experiment results so that in future sessions they can resume from what they have done previously as opposed to starting from scratch, (2) registered users get more detailed and more reliable prediction results using larger number of predictors and their fusions, and (3) registered users get to be notified with the personalized information on new improvements, hints and tips of the MFG system either when they log in to the system next time or via email. Registration requires user to enter their

email address, a password for log in, and other needed information (e.g., professional and socio-demographic characteristics).

Once a user logs into MFG, they are provided with two options: either to start a new scenario (a brand new movie prediction), or to look at the previous scenarios (movie predictions done previously and saved on the MFG database under that user). If a new scenario is selected, user is then given the parameter selection/entry interface (Fig. A2). Once all parameters are entered, and the forecast button is clicked, MFG initiates a request for prediction to all available Web services by sending the requisite data and parameters in XML. When the forecasts are returned by the Web services (also in XML), MFG initiates a request to the fusion Web service (FWS). The FWS combines the forecasts into an aggregated forecast and returns a value. MFG then presents these results both in textual (in the form of a performance class based on prediction dollar values) and in graphical (in the form of bar-chart) formats (Fig. A3). Once the forecast results are presented to the user, the

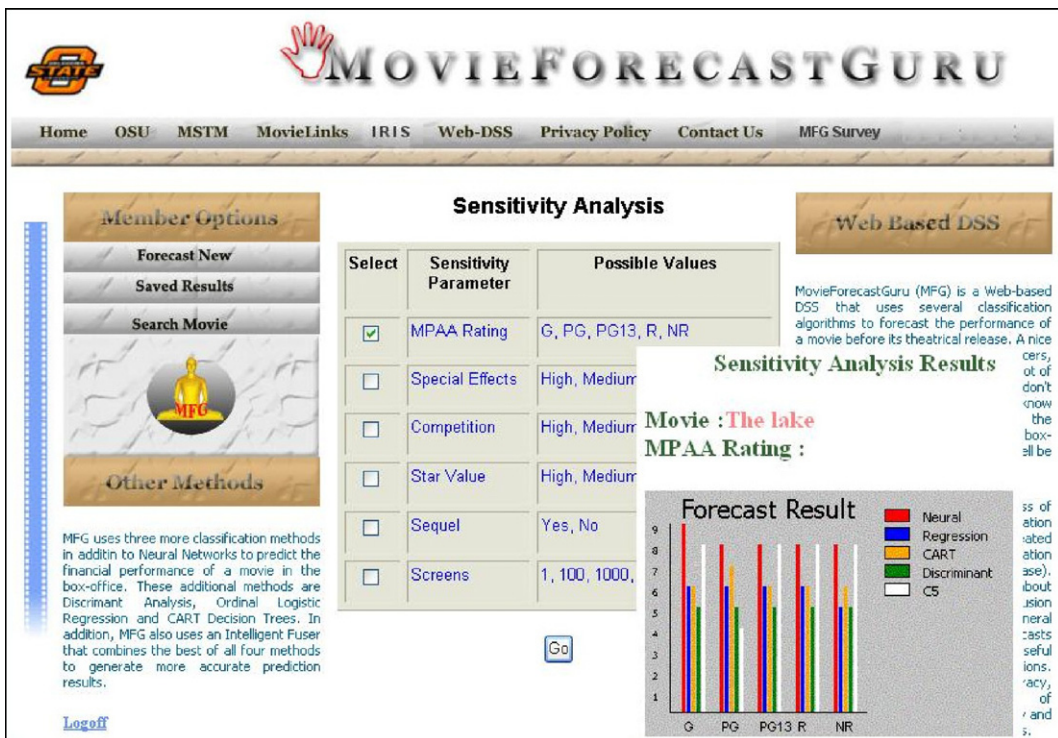


Fig. A4. Sensitivity analysis results interface.

user has the options of (1) terminating the interaction with MFG by logging out, (2) starting another new prediction scenario, (3) going back and changing a parameter value and rerun the prediction, and (4) conducting sensitivity analysis on the currently presented prediction scenario. If the sensitivity analysis is selected, then the user is presented with all decision variables and their possible values. The user is expected to choose one or more of these decision variables on which to conduct sensitivity analysis. MFG generates all possible combinations of scenarios from the selected list of decision variables and their possible values by repeating the process described above.

The results of the sensitivity analysis are also presented in both textual and in graphical formats (Fig. A4). The goal of the sensitivity analysis results is to show the user whether the dependent variable (financial success of a movie scenario) is sensitive to the changing values of one or more of the selected decision variables. For instance, it would show the user, for a specific movie scenario, whether the star value of an actor or actress plays a critical role in determining the financial success of that movie. If it does not, then the user might choose to use low star valued actors/actress (say a one million dollars actor versus twenty-five million dollars actor) and still expect to achieve similar box-office receipts.

If the user chooses to start from the previously ran movie scenarios, then s/he is provided with a list of saved experiments grouped by movie names. A movie can have more than one experiment associated with it, each having a different combination of parameter values. From the provided list, the user can choose any movie and change some of the parameter values. Once a prediction result is obtained, the user has the option of either saving it to the database or to discard it.

References

- [1] J.S. Armstrong, Combining forecasts, in: J.S. Armstrong (Ed.), Principles of Forecasting, Kluwer Academic Publishers, Norwell, MA, 2002, pp. 418–439.
- [2] T. Arora, Web-based analytics for the plant, Manufacturing Systems 18 (2000) 76–83.
- [3] AURORA, Advanced Models, Applications and Software Systems for High Performance Computing, URL: <http://www.univie.ac.at/sor/aurora6> (Vienna University, Vienna, Austria, 2004).
- [4] R. Batchelor, P. Dua, Forecaster diversity and the benefits of combining forecasts, Management Science 41 (1995) 68–75.
- [5] B. Bradway, R. Purchia, Leveraging customer relationships, Bank Systems and Technology 37 (2000) 31–39.
- [6] L. Breiman, J.H. Friedman, R.A. Olshen, C.J. Stone, Classification and Regression Trees, Wadsworth & Brooks/Cole Advanced Books and Software, Monterey, CA, 1984.
- [7] V. Catrini, No catch-22 here, Health Management Technology 23 (2002) 24–29.
- [8] C.W. Chase Jr., Composite forecasting: combining forecasts for improved accuracy, Journal of Business Forecasting Methods & Systems 19 (2000) 2–22.
- [9] F.D. Davis, Perceived usefulness, perceived ease of use, and user acceptance of information technology, MIS Quarterly 13 (3) (1989) 319–340.
- [10] W.H. DeLone, E.R. McLean, Information system success: the quest for the dependent variable, Information Systems Research 3 (1) (1992) 60–95.
- [11] I. De Silva, Consumer selection of motion pictures, in: B. Litman (Ed.), The Motion Picture Mega-Industry, Allyn & Bacon Publishing, Inc., Boston, MA, 1998.
- [12] J. Dong, H.S. Du, S. Wang, K. Chen, X. Deng, A Framework of Web-based Decision Support System for Portfolio Selection with OLAP and PVM, Decision Support Systems 37 (3) (2004) 367–376.
- [13] J. Dougherty, R. Kohavi, M. Sahami, Supervised and unsupervised discretization of continuous features, Proceedings of 12th International Conference on Machine Learning, Morgan Kaufmann, Los Altos, CA, 1995, pp. 194–202.
- [14] DSSResources, Decision Support Systems Resources by Dan Power, URL: <http://www.dssresources.com> (2004).
- [15] J. Eliashberg, J.J. Junker, M.S. Sawhney, B. Wierenga, MOVIEMOD: an implementable decision support system for prerelease market evaluation of motion pictures, Marketing Science 19 (3) (2000) 226–243.
- [16] J. Eliashberg, S. Swami, C.B. Weinberg, B. Wierenga, Implementing and evaluating silverscreener: a marketing management support system for movie exhibitors, Interfaces 31 (3) (2001) 108–127.
- [17] R.A. Fisher, The use of multiple measurements in taxonomic problems, Eugen 7 (1936) 179–188.
- [18] O. Gunter, R. Muller, P. Schmidt, H. Bhargava, R. Krishnan, MMM: a Web-based system for sharing statistical computing modules, IEEE Computer (1997 (May–June)) 59–68.
- [19] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, Springer-Verlag, New York, NY, 2001.
- [20] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice Hall, New Jersey, 1998.
- [21] K. Hornik, M. Stinchcombe, H. White, Universal approximation of an unknown mapping and its derivatives using multilayer feedforward network, Neural Networks 3 (1990) 359–366.
- [22] G.Q. Huang, K.L. Mak, Design for manufacture and assembly on the internet, Computers in Industry 38 (1) (1999) 17–28.

- [23] R. Kohli, F. Piontek, T. Ellington, T. VanOsdol, M. Shepard, G. Brazel, Managing customer relationships through e-business decision support applications: a case of hospital-physician collaboration, *Decision Support Systems* 32 (2) (2001) 171–179.
- [24] G.L. Kovacs, P. Paganelli, A planning and management infrastructure for large, complex, distributed projects-beyond ERP and SCM, *Computers in Industry* 51 (2) (2003) 165–173.
- [25] D. Li, A. McKay, A. de Pennington, C. Barnes, A Web-based tool and a heuristic method for cooperation of manufacturing supply chain decisions, *Journal of Intelligent Manufacturing* 12 (5–6) (2001) 342–433.
- [26] B.R. Litman, Predicting success of theatrical movies: an empirical study, *Journal of Popular Culture* 16 (9) (1983) 159–175.
- [27] B.R. Litman, H. Ahn, Predicting financial success of motion pictures, in: B.R. Litman (Ed.), *The Motion Picture Mega-Industry*, Allyn & Bacon Publishing, Inc., Boston, MA, 1998.
- [28] B.R. Litman, A. Kohl, Predicting financial success of motion pictures: the 80's experience, *The Journal of Media Economics* 2 (1) (1989) 35–50.
- [29] H. Liu, H. Farhad, T.L. Chew, M. Dash, Discretization: an enabling technique, *Data Mining and Knowledge Discovery* 6 (2002) 393–423.
- [30] W.K. McHenry, Using knowledge management to reform the Russian criminal procedural codex, *Decision Support Systems* 34 (3) (2003) 339–352.
- [31] Microsoft. MSDN Library (Visual Studio .NET 2003).
- [32] R. Neelamegham, P. Chintagunta, A Bayesian model to forecast new product performance in domestic and international markets, *Marketing Science* 18 (2) (1999) 115–136.
- [33] NEOS, A Web-based Optimization Solver, URL: <http://www-neos.mcs.anl.gov/neos> (Argonne National Lab and Northwestern University 2004).
- [34] Not Disclosed. In order to comply with the double-blind review process of the manuscripts, the specifics of these references are not revealed.
- [35] G.C. O'Connor, B. O'Keefe, Viewing the web as a marketplace: the case of small companies, *Decision Support Systems* 21 (1997) 171–183.
- [36] OOF, Open Optimization Framework, URL: <http://www.doc.ic.ac.uk/~oce/elsinore/index.htm> (Imperial College, London, UK. 2003).
- [37] OSP, Optimisation Service Provider, URL: <http://www.osp-craft.com> (European Union CRAFT Project, OptiRisk Systems 2004).
- [38] D.J. Power, *Decision Support Systems: Concepts and Resources for Managers*, Quorum Books, Westport, CT, 2002.
- [39] D.J. Power, S. Kaparathi, Building Web-based Decision Support Systems, *Studies in Informatics and Control* 11 (4) (2002) 291–302.
- [40] J. Quinlan, Induction of decision trees, *Machine Learning* 1 (1986) 81–106.
- [41] J. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann, San Mateo, CA, 1993.
- [42] S.A. Ravid, Information, blockbusters, and stars: a study of the film industry, *Journal of Business* 72 (4) (1999) 463–492.
- [43] E. Ruiz, F.H. Mieto, A note on linear combination of predictors, *Statistics & Probability Letters* 47 (2000) 351–356.
- [44] M.S. Sawhney, J. Eliashberg, A parsimonious model for forecasting gross box-office revenues of motion pictures, *Marketing Science* 15 (2) (1996) 113–131.
- [45] H. Schildt, *C#, A Beginner's Guide*, Osborne/McGraw-Hill, Berkeley, CA, 2001.
- [46] J.P. Shim, M. Warkentin, J.F. Courtney, D.J. Power, R. Sharda, C. Carlsson, Past, present, and future of decision support technology, *Decision Support Systems* 33 (2002) 111–126.
- [47] ShowBiz Data, Inc. URL: <http://www.showbizdata.com> (Los Angeles, CA. 2003).
- [48] H.A. Simon, *The Sciences of the Artificial*, 3rd edition, The MIT Press, Cambridge, MA, 1995.
- [49] S. Sochay, Predicting the performance of motion pictures, *The Journal of Media Economics* 7 (4) (1994) 1–20.
- [50] R.P. Sundarraj, A Web-based AHP Approach to Standardize the Process of Managing Service Contracts, *Decision Support Systems* 37 (3) (2004) 343–365.
- [51] J. Valenti, Motion Pictures and Their Impact on Society in the Year 2000, A Speech Given at the Midwest Research Institute, Kansas City, April 25 (1978).
- [52] T.M. Vitolo, R.J. Vance, STEP-UP: a decision-support system for transforming the dislocated US defense workforce, *Interfaces* 32 (4) (2002) 75–88.
- [53] WEBOPT, Web-enabled Optimization Tools and Models, ASIA IT and C Project, URL: <http://www.webopt.org> (CARISMA 2004).
- [54] D.E. Woodin, Sentencing with gungweb: speculations on the value of rule-based systems in a world of badly drafted statutes, *Information and Communications Technology Law* 11 (1) (2002) 49–57.
- [55] J. Zeleznikow, Using Web-based decision support systems to improve access to justice, *Information and Communications Technology Law* 11 (1) (2002) 16–24.
- [56] F.S. Zufryden, Linking advertising to box office performance of new film releases: a marketing planning model, *Journal of Advertising Research* (1996 (July–August)) 29–41.



Dr. Dursun Delen is an assistant professor of management science and information systems in the Spears School of Business at Oklahoma State University. His research appeared in *Communications of the ACM*, *Computers in Industry*, *Computers and Operations Research*, *Intelligent Manufacturing Systems*, *Artificial Intelligence in Medicine*, among others. His research interests include decision support systems, enterprise engineering, artificial intelligence, data mining and knowledge management.



Dr. Ramesh Sharda is Director of the Institute for Research in Information Systems (IRIS) ConocoPhillips Chair of Management of Technology, and a Regents Professor of Management Science and Information Systems in the College of Business Administration at Oklahoma State University. His research has been published in major journals in management science and information systems including *Management Science*, *Information Systems Research*, *Decision*

Support Systems, *Interfaces*, *INFORMS Journal on Computing*, *Computers and Operations Research*, and many others. He serves on the editorial boards of journals such as the *INFORMS Journal on Computing*, *Decision Support Systems*, *Information Systems Frontiers*, and *OR/MS Today*. His research interests are in decision support systems, information systems support for collaborative applications, and technologies for managing information overload. Ramesh is also a cofounder of a company that produces virtual trade fairs, iTradeFair.com.

Mr. Prajeeb Kumar is a Senior dot Net developer for United Communications Group. He completed his Masters in Telecommunications Management from Oklahoma State University, Stillwater and bachelor's degree in Electronics and Telecommunications from National Institute of Technology, India. He has worked on neural networks and decision support systems. His current interests mainly focus on the security aspects of Internet applications.