Flight Tickets Booking Decision Support System

Final Project of Course INFSCI 2130/ ISSP 2240 -

Decision Analysis & Decision Support Systems

Chaoran Zhou, Yue Zhang, Jiayu Liu, Sanchuan Jin, Xi Zhang

1. Problem Description

1.1 Introduction

Nowadays with increasing number of airline companies and flights people have more and more options when taking a flight trip. However, sometimes it is very time consuming to figure out a best flight from a huge amount of airline options for a trip. In an actual flight booking decision the best choice doesn't only mean the one with lowest price, but also means short flight time, fewer stops, reasonable layover time and etc. A flight passenger often faces a tough decision problem how to select an airline flight ticket with multiple objectives.

1.2 Objectives

- Save money
- Save time:
  - less flight time
  - less layover time

1.3 Conflicting Objectives

The two objectives saving money and saving time are conflicting. A ticket with lower price usually has longer flight time and longer layover time.

1.4 Main Considerations

To solve the problem we take three main considerations here:

1. Ticket price
2. Flight time
3. Layover time

2. Utility Model

2.1 Model Description

We built a utility model to describe the flight ticket booking decision as the following Figure 1. In this model we want to figure out what contribute to the utility value of a flight passenger and how they influence a passenger's utility value. Table 1 just following Figure 1 gives description for each node in the model.

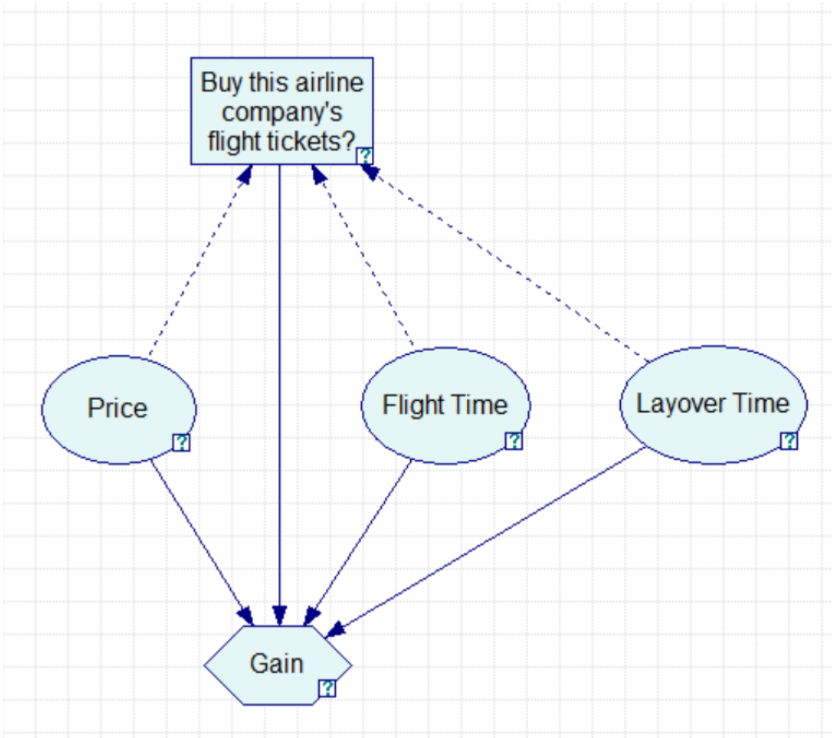


Figure 1 Influence Diagram of the Flight Booking Decision Model

Table 1 Description for Each Node in the Model

Node Type	Node Name	Node Description
Decision	Buy this airline company's flight tickets?	2 choices: • Buy • Do Not Buy
Chance	Price	The price of an airline company ticket for this flight trip with 3 states: • High • Middle • Low
Chance	Flight Time	A total time of all flights for this trip (layover time is excluded) with 3 states: • Long • Middle • Short
Chance	Layover Time	A total time that a passenger should stay in airports to transit from one airplane to another with 3 states: • Long • Middle • Short
Value	Gain	Expected utility if a passenger buys one airline company's flight tickets

2.2 Quantification of Uncertainties

Our system collects all tickets' prices from all airline companies. First, our system calculates the difference between the highest price and the lowest one. Then, the system divides the difference into three equal intervals as high, middle, and short. Finally, the system figures out prices' distribution of each company in the three intervals.

By the same method the system can get each company's tickets distribution of flight time and layover time on long, middle and short intervals.

2.3 Quantification of Preferences

In current model we assume that a passenger has same preferences among price, flight time and layover time, that is, the three uncertainties have the equal weights in our model now. Preference parameters see the following Table 2.

Table 2 Preference Parameters

Price	Flight Time	Layover Time	Preference Parameter
High	Long	Long	-100
Middle	Middle	Middle	0
Low	Short	Short	+100

3. Data Collecting & Processing

In this project we first analyzed the original URL of Expedia.com flight search, and then found out the parameters, which the URL consists of. After a user inputs his/her flight trip requirements into our system, the system generates an Expedia format URL with this user's flight requirement parameters and then sends this URL to the Expedia server.

Expedia server feeds back all flight search results, but the results' more details are in AJAX, an asynchronous loading format, which can't be parsed by usual approaches, such as HTMLparse, that only work for static webpages and are not able to request JS or JSP files from server sides. To solve this problem we use Selenium, an automatic browser, which has the support from some large browser vendors. Selenium is primarily used for testing purposes to automatically run web applications. However, the usage of Selenium is not only the abovementioned. It can be also used to automatize some boring web-based administration tasks. Selenium is integrated into our System as an API to run Firefox browser to simulate the AJAX asynchronous uploading process. And then our system can obtain the dynamically uploaded webpage code.

After parsing the entire DOM tree the system retrieves all useful flight data. All flight data is stored in Hash Maps after necessary pre-processes and calculations.

4. How the System Runs

A user types in his/her departure and destination and also select his/her departure date in the system's input interface as the following Figure 2. All input information is encoded into an Expedia flight search request URL, and then the URL is sent to Expedia server after the user clicks on submit button.

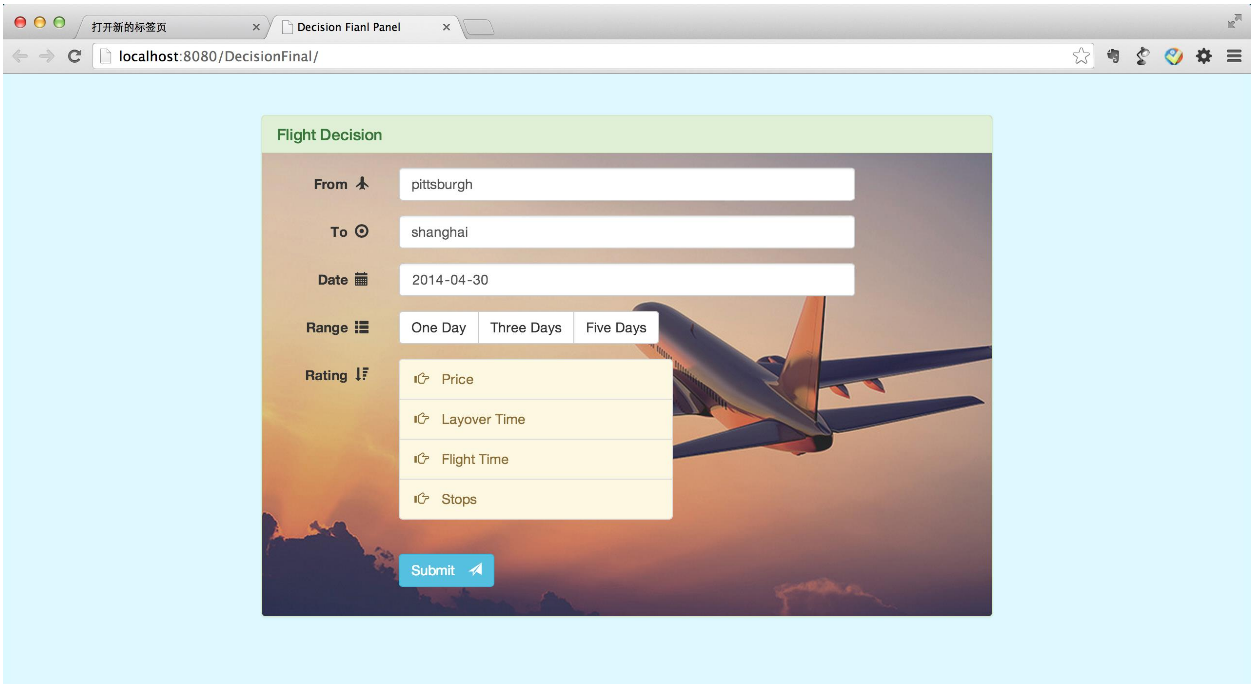
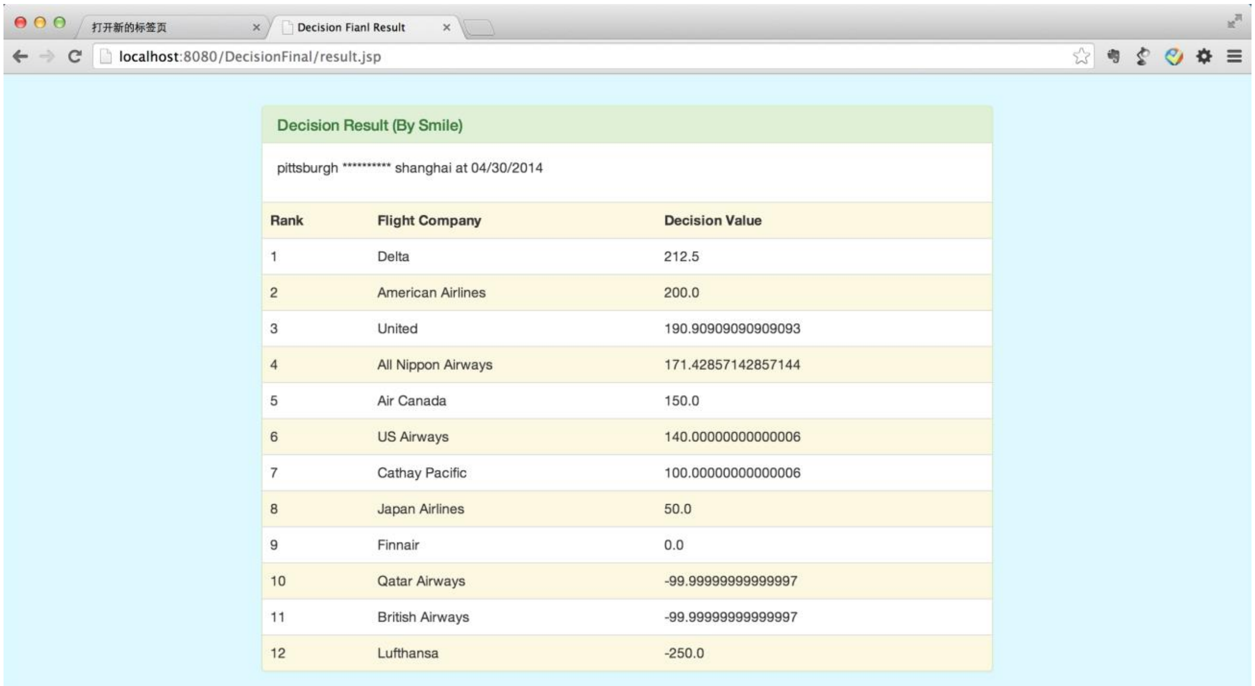


Figure 2 Input Interface of the System

Expedia server feeds search results webpage back. Our system traverses the dynamic webpage to extract all flights' data. Based on our utility model the system calculates an expected utility for each airline company's flight tickets. Finally our system outputs a rank and a utility value (decision value) for all airline companies (flight companies) as following Figure 3. If we check this flight search results in Expedia (please download ExpediaResults.png and ExpediaResultsDetails.png), we can find that our airline company ranking results correspond to the Expedia's flight ticket search results.

Note:

In our system a flight ticket, which has different airline companies providing different parts of a whole flight trip, is set as default to be sold by the first part airline company.



Decision Result (By Smile)

pittsburgh ********** shanghai at 04/30/2014

Rank	Flight Company	Decision Value
1	Delta	212.5
2	American Airlines	200.0
3	United	190.90909090909093
4	All Nippon Airways	171.42857142857144
5	Air Canada	150.0
6	US Airways	140.00000000000006
7	Cathay Pacific	100.00000000000006
8	Japan Airlines	50.0
9	Finnair	0.0
10	Qatar Airways	-99.99999999999997
11	British Airways	-99.99999999999997
12	Lufthansa	-250.0

Figure 3 Results Page of the System

5. Future Work

5.1 Rank Flight Tickets Rather Than Airline Companies

The original intention of developing current system is to research and compare different airline companies’ utilities for a passenger’s flight trip based on the companies’ all tickets. To make our system more useful and helpful for flight passenger users, the future system should grade and rank flight tickets rather than airline companies.

Expedia.com is one of the most popular flight booking website now. It sorts its tickets for a flight trip by four individual ways:

- Price: low to high, the default sorting
- Duration (total time of flight time and layover time): short to long
- Departure: morning to night in the selected departure date
- Arrival: early to late

Not any combination of the above-mentioned four ways is used to do the sorting work.

Expedia also marks one ticket, whose price is relatively low and flight duration is relatively short simultaneously, as the “Best Value” among all tickets for a trip.

However, what if tickets are sorted by utility value generated from a utility model including multiple uncertainties? This kind of ranking should be more user-friendly than existing ones. Our future work will majorly focus on this new ranking method.

5.2 Consider More Uncertainties

In our current utility model only three uncertainties (price, flight time, and layover time) are taken into consideration. To help flight passengers make better decisions more uncertainties can be added into the model, such as:

- Stops amount in a flight duration
- Flight On Time Rate
- Flight Service Rating

5.3 Better Model Parameters Setting

5.3.1 Non-Monotonic Preferences

In most cases lower price and shorter flight time are more preferred by passengers, but not all uncertainties' preferences have monotonicity. For example, short layover time in an airport doesn't mean good for a passenger, who has a terminal transit in a stop airport or has entry and exit procedures in an international airport as a stop during an international flight trip. Another example is that too early or too late departure or arrival time in a day could not be convenient for a flight passenger.

5.3.2 Further Optimization

To optimize Flight On Time Rate we can use time series analysis to find out some patterns, such as seasonal patterns, to avoid some low On Time Rate flights to suit a passenger's preference for keeping one's own schedule when he/she is booking a flight tickets.

5.3.3 Priorities of Uncertainties

A challenge in the future is the priority of uncertainties. In our current utility model every of the three uncertainties has the same priority. However, a real passenger may have his/her own priorities for the uncertainties. In our current input interface we design a rating panel, where a user can rank the uncertainties by herself/himself, as the following Figure 4. We haven't implemented the ranked uncertainties in the current model, because how to exactly set uncertainties' weights is still a tough problem. Machine Learning may be used to solve this problem in the future.

Figure 4 Uncertainties Rating Panel

5.4 Wider Departure Date Range Selection

We have designed a departure date range option on the current system input interface as the following Figure 5. In the future system if a user's departure date is flexible, he/she can search the flight tickets within one/three/five days after the selected departure date. So in a wider departure date range a user should have more opportunities to find tickets with better utility values recommended by our flight tickets booking decision support system.

Figure 5 Departure Date Range Option

6. Project Development Tools

- **GeNIe -** a development environment for building graphical decision-theoretic models, source: https://dslpitt.org/genie/wiki/Introduction:_GeNIe
- **jSMILE** - a platform independent* library of Java classes for reasoning in graphical probabilistic models, such as Bayesian networks and influence diagrams, source: https://dslpitt.org/genie/wiki/Introduction_to_jSMILE
- **Selenium** - Web Browser Automation, source: http://www.seleniumhq.org/

7. Project Task Responsibilities

Project Management:	Chaoran Zhou
Utility Model Designing:	Chaoran Zhou Yue Zhang Xi Zhang
Model Setting, Developing & Programming:	Chaoran Zhou
Data Collection & Process Developing:	Sanchuan Jin
User Interface Designing & Developing:	Jiayu Liu
Reporting:	Chaoran Zhou Sanchuan Jin Jiayu Liu Yue Zhang Xi Zhang