Developing Actionable Trading Strategies for Trading Agents

Chengqi Zhang
Director
Centre for Quantum Computation and Intelligent Systems (QCIS),
University of Technology, Sydney, Australia
Key Ideas in this talk

- One scenario
  - If an agent is given one million dollars to trade, which stock, when, buy or sell, how many shares should be traded?

- Key points
  - Domain Knowledge is a key to developing actionable Trading Agent
  - Individual Smart trading strategy is important
  - Trading strategy integration
  - This talk will focus on how to develop actionable trading strategies
Relationships between the profits and strategies with parameters
Contents

- About QCIS Centre
- What is trading agent
- What is trading strategy
- Actionable trading agent/strategy
- Trading strategy optimization
- Enhancing trading strategy
- Multi-trading strategy integration
- F-Trade: support smart trading
- Conclusions
About QCIS Centre

- **Name**
  - Centre for Quantum Computation and Intelligent Systems (QCIS)

- **People**
  - *25* researchers include 6 Profs, 5 A/Profs, 4 Senior Lecturers, 4 Lecturers, 6 Postdocs, and 40 Research Students

- **Achievements**
  - *11* ARC Grants in 2009 in life include 8 ARC DP and 3 ARC LP (AU$1,000,000+)
  - Industry grants in 2009 (AU$500,000+)
  - One of the leading research centres in Australia
Research Laboratories

- Quantum Computation Laboratory
- Data Sciences and Knowledge Discovery Laboratory
- Decision Systems and e-Service Intelligence Laboratory
- Knowledge Infrastructure Laboratory
- Innovation and Enterprise Research Laboratory
National Grants in Knowledge Discovery Lab (2009)

1. Domain Driven Data Mining (ARC DP 2007-2009)
2. Data Mining of Activity Transactions to Strengthen Debt Prevention (ARC LP 2007-2009)
3. Discovering Activity Patterns Driven by High Impacts in Heterogeneous and Imbalanced Data (ARC DP 2009-2011)
4. Multiple Data Source Discovery: Group Interaction Approach (ARC DP 2009-2013)
5. Pattern Analysis and Risk Control of E-Commerce Transactions to Secure Online Payments (ARC LP 2009-2011)
Industry Grants in Knowledge Discovery Lab (2009)

- Applications
  - Stock Market Surveillance & Trading
  - Centrelink Debt Prevention
  - Fraud management for on-line e-payments
  - HCF (Medical Insurance Fraud detection)
Contents

- About QCIS Centre
- What is trading agent
- What is trading strategy
- Actionable trading agent/strategy
- Trading strategy optimization
- Enhancing trading strategy
- Multi-trading strategy integration
- F-Trade: support smart trading
- Conclusions
What is Trading Agent?

- Automated decision
- Smart decision
- Workable decision

>>> Actionable trading strategies
What is Trading Strategy

- In finance, a trading strategy (see also trading system) is a predefined set of rules for making trading decisions. (Wikipedia)
- A trading strategy indicates *when* a trading agent can take *what* trading actions under certain market *situation*.
What is Trading Strategy

- Trading strategy design problem
  - trading strategy set $\Omega = \langle T, B, P, V, I \rangle$
    - time $T = \{ t_1, t_2, \ldots, t_m \}$
    - behavior $B = \{ \text{buy}, \text{sell}, \text{hold} \}$
    - price $P = \{ p_1, p_2, \ldots, p_m \}$
    - volume $V = \{ v_1, v_2, \ldots, v_m \}$
    - instrument $I = \{ i_1, i_2, \ldots, i_m \}$
  - Our goal: actionable trading strategy set $\Omega'$ ($\Omega' \subseteq \Omega$)
    $$\Omega' = \{ (\omega, \delta) | \omega \in \Omega, \delta \in \{(\delta^k, a) \} \}$$
    - trading agent $a$ ($a \in A$)
    - $\omega$: optimal strategy instance
    - $\delta$: all constraint instances
What is Trading Strategy

- An trading strategy example

**TRADING STRAGE 1: A generic strategy** FR(δ)

At time point $t$, get $\text{high}(t)$ and $\text{low}(t)$

IF $\text{price}(t-1) > \text{high}(t-1)$

$\text{high}(t) = \text{price}(t-1)$

ELSE

$\text{high}(t) = \text{high}(t-1)$

IF $\text{price}(t-1) < \text{low}(t-1)$

$\text{low}(t) = \text{price}(t-1)$

ELSE

$\text{low}(t) = \text{low}(t-1)$

Generate trading signals

IF $\text{price}(t) < \text{high}(t)^*(1 - \delta)$

Generate SELL signal

IF $\text{price}(t) > \text{low}(t)^*(1 + \delta)$

Generate BUY signal
Actionable trading agent/strategy

- Actionable trading strategy
- Trading strategy optimization
- Trading strategy enhancement
- Trading strategy integration
- Trading support system
Actionable trading strategy

\[
\begin{align*}
\text{tech\_int}(t, b, p, v, i) & \rightarrow \max[\text{tech\_int}()] \\
\text{biz\_int}(t, b, p, v, i) & \rightarrow \max[\text{biz\_int}()] \\
\end{align*}
\]

s.t.

\[
\begin{align*}
\Omega' &= \{e_1, ..., e_n\} \\
\Omega' &\subset \Omega \\
m &> n \\
\alpha_z &= \sum b_i \times p_i \times v_i \\
\beta_z &= \sum |b_i| \times \beta_i \times p_i \times v_i \\
SR &= (R_p - R_f) / \sigma_p \\
TR &= \frac{\sum_{i=1}^{u} \text{AskPrice}_i \times \text{AskVolume}_i - \sum_{j=1}^{v} \text{BidPrice}_j \times \text{BidVolume}_j}{\text{TotalInvestment}} \\
IR &= \left( \sum_{i=1}^{n} (\text{Index}_{i+1} - \text{Index}_i) / \text{Index}_i \right) / n
\end{align*}
\]
Trading Strategy Optimization

- Evolutionary trading agent
  - Modeling roles
    - Crossover
    - Mutation
    - ...

Role $R_{\text{mutateCandidateStrategies}}$
Statement Mutation is a process that parts of a chromosome are to be changed. This role determines to what extent the parts of a chromosome in a trading agent are to be mutated. The extent is the mutation rate.
Agent $A_{\text{EvolutionaryAgent}}$
Agent $A_{\text{UserAgent}}$
Agent $A_{\text{StrategyAgent}}$
Agent $A_{\text{CoordinatorAgent}}$
Attribute $aa:A_{\text{EvolutionaryAgent}}$
Attribute constant mutRate:MutationRate
Attribute $\text{parad}:/A_{\text{InParameters}}$
Attribute $aa:A_{\text{UserAgent}}$
Attribute $asa:A_{\text{StrategyAgent}}$
Attribute constant strid:asa
Attribute $aa:A_{\text{CoordinatorAgent}}$
Protocol receiveStrategyMutationRequest
Protocol checkStrategyAgentValidity
Protocol openMutationSettingInterface
Protocol submitStrategyMutationRequest
Protocol returnStrategyMutationResponse
Responsibilities
Liveness
\[ \text{strid}.aca.checkStrategyAgentValidity() \rightarrow \text{asa}.openMutationSettingInterface(asa, asa.parad[]): \]
\[ \rightarrow \text{aca}.receiveStrategyMutationRequest(aca) \rightarrow \text{aca}.submitStrategyMutationRequest(aca) \rightarrow \text{aca}.executeStrategyMutation(aca, mutRate, aca) \rightarrow \text{aca}.returnStrategyMutationResponse(aca, aca) \]
Safety (Invariant) $0 < \text{mutRate} < 1.0$
Trading strategy optimization

Develop optimized trading strategies for trading agents:
- Optimized trading strategies

Checking business performance:
- Actionability of trading strategies
Relationships between the profits and strategies with parameters
Enhancing Trading Strategy

- Domain factors

\[ M = \{I, A, O, T, R, E\} \]
\[ \sum = \{\delta_i^k | c_i \in C, k \in N\} \]
\[ \Omega = \{(\omega, \delta) | \omega \in \Omega, \delta \in \{\delta_i^k, a\}, \delta_i^k \in \sum, a \in A\} \]

Table 1: Domain factors and its impact to actionability

<table>
<thead>
<tr>
<th>Organizational factors</th>
<th>Impact to actionability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traded instruments (I), such as stock or derivatives, (I = {\text{stock, option, feature, ...}})</td>
<td>Varying instruments determine different data, analytical methods and objectives</td>
</tr>
<tr>
<td>Market participants (A), (A = {\text{broker, market maker, mutual funds, ...}})</td>
<td>Traders have the final right to evaluate and deploy discovered trading evidence to their advantage</td>
</tr>
<tr>
<td>Orderbook forms (O), (O = {\text{limit, market, quote, block, stop}})</td>
<td>Order type determines what data set (e.g., orderbook) to be mined, as well as particular business interestingness</td>
</tr>
<tr>
<td>Trading session, indicated by timeframe (T) showing whether a market includes call market or continuous session</td>
<td>Setting up the focusing session can prune order transactions</td>
</tr>
<tr>
<td>Market rules (R), e.g., restrictions on order execution defined by exchange</td>
<td>They determine pattern validity of discovered trading patterns when deployed</td>
</tr>
<tr>
<td>Execution system (E), e.g., a trading engine is order or quote-driven</td>
<td>It limits pattern type and deployment manner after migrated to a real trading system</td>
</tr>
</tbody>
</table>
Enhancing trading strategies

- Based on a basic strategy, say FR(δ)
- Add domain specific factors
- For instance,

FR(δ)

FR(t, δ_H, δ_L, h, d)

TRADING STRATEGY 2: An enhanced FR(t, δ_H, δ_L, h, d)
At time point t, get high(t) and low(t)
IF price(t-1) > high(t-1)
   high(t) = price(t-1)
ELSE
   high(t) = high(t-1)
IF price(t-1) < low(t-1)
   low(t) = price(t-1)
ELSE
   low(t) = low(t-1)
Generate trading signals
IF price(t) < high(t)*(1 - δ_H)
   Generate SELL signal
   IF position(t-1) <> 0 & hold(t-1) = h
      position(t) = 1
IF price(t) > low(t)*(1 + δ_L)
   Generate BUY signal
   IF position(t-1) <> 0 & hold(t-1) = h
      position(t) = -1
Results

- Enhancing trading strategies
  - Filter Rule Enhanced: FR(δ,h)
  - FR(δ,h) can greatly beat FR(δ)

<table>
<thead>
<tr>
<th>Trade</th>
<th>Trade</th>
<th>Maximum</th>
<th>Minimum</th>
<th>Net Long</th>
<th>Net Short</th>
<th>trades</th>
<th>Position</th>
<th>Profit</th>
<th>Payoff</th>
<th>Cumulative</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/2/2003</td>
<td>3035</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1/6/2003</td>
<td>3075</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1/10/2003</td>
<td>3025</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Figure 2.** Some results of enhanced trading strategy FR
Results

- Enhancing trading strategies
  - Filter Rule Enhanced: FR(δ,h)
  - FR(δ,h) can greatly beat FR(δ)

Figure 3. Performance comparison between base and enhanced trading strategies
(a) Sharpe ratio of a generic MA($4,19$)
(b) Sharpe ratio of an actionable rule MA($t,4,19,0.033,0,0,0$)

Figure 5: Improved business interestingness by mining in-depth trading rules
Mining Trading Strategy - stock Pair

- An example
Mining Trading Strategy

- An example

---

**Trade Rule-Stock Pairs**: TRSP\( (T, R, S, \rho_\theta, sr_\theta, r_\theta) \)

**Input**: a set of historical intraday orderbook transactions \( T \), a set of trading rules \( R \), a set of stocks \( S \), a coefficient threshold \( \rho_\theta \), a sharpe ratio threshold \( sr_\theta \), and a return threshold \( r_\theta \)

**Output**: Fuzzily ranked trading rule-stock pairs

**Interestingness**: fuzzy ranking coefficient \( \rho \), and sharpe ratio \( SR \)

**Constraints**: intraday data, market niche \( M \)=\{ASX stocks, any traders, market orderbook, continuous session, ASX order, none\}

**Method**:

1. Given a stock \( S_i \) and a type of trading rule \( R_j \), identify the in-depth rules \( r_{jm} \) \((m = 0,1,\ldots)\) for the stock in \( T \);
2. Identify all in-depth rules \( r_{jm} \) \((i =0,1,\ldots; j =0,1,\ldots; m \) = 0,1,\ldots\) for all stocks \( S_i \) \((i =0,1,\ldots)\) and all types of trading rules \( R_j \) \((j =0,1,\ldots)\);
3. Fuzzily aggregate the rule set \( r_{jm} \), and rank them to generate a fuzzily “optimal” rule for a given stock;
4. Generate fuzzy “optimal” rule-stock pairs \( (s_i, r_j) \) \((i =0,1,\ldots; j =0,1,\ldots)\) in terms of stocks and trading rules;
5. Evaluate rule-stock pairs \( (s_i, r_j) \) in terms of technical and business interestingness measures respectively;
6. Aggregate and fuzzily rank rule-stock pairs \( (s_i, r_j) \) in terms of ranking coefficient and business performance;
7. Evaluate and recommend highly-ranking rule-stock pairs for trading support.
---
Profit to investment

![Graph showing profit to investment](image-url)
Fig. 6. Monthly return $TR$ of top 10% rule-stock pairs (Technical interest only vs. fuzzy two-way significance vs. equally weighted two-way significance)
Multi-trading strategy integration

Table 1. Trading strategy base

<table>
<thead>
<tr>
<th>Class</th>
<th>Types in a class</th>
</tr>
</thead>
<tbody>
<tr>
<td>FR</td>
<td>FR-X, FR-XC, FR-XY, FR-XE</td>
</tr>
<tr>
<td>MA</td>
<td>MA-MN, MA-BMN, MA-CMN, MA-DMN</td>
</tr>
<tr>
<td>CB</td>
<td>CB-NXC, CB-NXBC</td>
</tr>
<tr>
<td>SR</td>
<td>SR-N, SR-NB, SR-NC, SR-NBC, SR-NDC</td>
</tr>
<tr>
<td>OBV</td>
<td>OBV-MN, OBV-B, OBV-C, OBV-D</td>
</tr>
</tbody>
</table>
- Evolutionary trading agent searches golden strategy for each class
- Golden trading agents negotiate for the local best
- Coordinator agent monitors for global best
- Coordinator agent selects and aggregates positions for all golden strategies
- Collaborative agent trades all selected golden strategies
Figure 1. Multi-strategy integration for trading agents
Table 3. Data partition excerpt

<table>
<thead>
<tr>
<th></th>
<th>starting</th>
<th>Train end</th>
<th>Deploy end</th>
</tr>
</thead>
<tbody>
<tr>
<td>Window 1</td>
<td>1/1/1997</td>
<td>30/12/1998</td>
<td>30/12/2000</td>
</tr>
<tr>
<td>Window 2</td>
<td>1/1/1998</td>
<td>30/12/1999</td>
<td>30/12/2001</td>
</tr>
<tr>
<td>Window 3</td>
<td>1/1/1999</td>
<td>30/12/2000</td>
<td>30/12/2002</td>
</tr>
</tbody>
</table>
Outputs

Table 5. Maximal benefits to parameter combinations (excerpt) 
(Data: 2003; Market: ASX; Strategy: MA-BMN)

<table>
<thead>
<tr>
<th>Parameter combinations</th>
<th>Benefit ($)</th>
<th>Cost ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>n = 2; x = 0.010; c = 25; b = 0.001</td>
<td>32350</td>
<td>394.25</td>
</tr>
<tr>
<td>n = 5; x = 0.010; c = 75; b = 0.015</td>
<td>18200</td>
<td>268</td>
</tr>
<tr>
<td>n = 5; x = 0.010; c = 50; b = 0.001</td>
<td>16900</td>
<td>241.75</td>
</tr>
<tr>
<td>n = 2; x = 0.015; c = 25; b = 0.005</td>
<td>16550</td>
<td>256.5</td>
</tr>
<tr>
<td>n = 2; x = 0.020; c = 25; b = 0.001</td>
<td>12725</td>
<td>214.5</td>
</tr>
</tbody>
</table>

Table 4. Output excerpt of a trading strategy 
(Strategy: MA-BMN; Data: 2004)

<table>
<thead>
<tr>
<th>Date</th>
<th>Price</th>
<th>Sell</th>
<th>Buy</th>
<th>Position</th>
<th>($) Benefit</th>
<th>($) Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>2004-8-16</td>
<td>3466</td>
<td>-1</td>
<td>0</td>
<td>-1</td>
<td>9200</td>
<td>103</td>
</tr>
<tr>
<td>2004-8-17</td>
<td>3480</td>
<td>-1</td>
<td>0</td>
<td>-1</td>
<td>8850</td>
<td>106.5</td>
</tr>
<tr>
<td>2004-8-18</td>
<td>3472</td>
<td>-1</td>
<td>0</td>
<td>-1</td>
<td>9150</td>
<td>108.5</td>
</tr>
<tr>
<td>2004-8-19</td>
<td>3481</td>
<td>-1</td>
<td>0</td>
<td>-1</td>
<td>8825</td>
<td>110.75</td>
</tr>
<tr>
<td>2004-8-20</td>
<td>3494</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>8500</td>
<td>114</td>
</tr>
</tbody>
</table>

Table 6. Trading agent positions recommended by five trading strategy classes (excerpt) 
(Strategy class: MA, FR, CB, SR, OBV; Data: Hongkong; Year: 2006)

<table>
<thead>
<tr>
<th>Date</th>
<th>Position MA</th>
<th>Position FR</th>
<th>Position CB</th>
<th>Position SR</th>
<th>Position OBV</th>
</tr>
</thead>
<tbody>
<tr>
<td>2006-11-16</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2006-11-17</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2006-11-20</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2006-11-21</td>
<td>-1</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2006-11-22</td>
<td>-1</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Figure 2. Cumulative benefits of each trading strategies (Year: 2003-2006, Market: Hongkong, Strategies: MA, FR, CB, SR, OBV, and Integrative)

Figure 3. Cumulative cost/benefit ratio of each golden trading strategies (Year: 2003-2006, Market: Hongkong, Strategies: MA, FR, CB, SR, OBV, and Integrative)
F-Trade: Support Smart Trading

- Support Trading,
  - e.g., identifying better trading rules
- Support Surveillance,
  - e.g., identifying exceptional trading behavior
- Support Data Mining,
  - e.g., developing actionable trading strategies
- Support Agents,
  - e.g., developing multi-trading agent learning
Organizational scheme

Users/CMCRC/Institutions
(Anybody, anytime, anywhere, from MAS, KDD & Finance areas and applications developers)

KDD Researchers
(Patterns discovery, optimization, actionability, knowledge management, …;
Finance, social security, …)

AAMAS Researchers
(Open complex agent systems, organization-oriented modeling, agent service-based computing, social intelligence, intelligence meta-synthesis, …)

Network
(Internet & LAN)

F-Trade
(www.f-trade.info
Open automated/human-cooperated enterprise and personalized service infrastructure)

Data Sources
(AC3 data including global markets; specific user data; etc.
Formats: FAV, ODBC, JDBC, OLEDB, CSV, etc.)
System environment

- Data
  - Global market orderbook data (tick-by-tick & daily)
  - AC3, CMCRC, SIRCA Ltd.

- Implementation
  - Web-based
  - Java, C, XML, SQL
  - Unix, Linux, Windows
  - App server (UTS) + database server (UTS) + data warehouse (AC3) + browsers

- Trading rules/strategies
  - Brokers/firms/financial researchers/data mining

- System history
Agent-based data mining infrastructure

- Software engineering of open complex agent systems
- OSOAD: Organization and Service Oriented Analysis and Design
  - Organizational abstraction
  - Organization-oriented analysis
  - Agent service-oriented design
- Agent service-based plug-n-play
  - Agent service-based system modules and services
  - Agent-based trading rules, DM algorithms
  - Remote data access, message passing, transactional processing, data sources
- Agent ontology-based management
  - Ontology for managing modules, algorithms, data sources, users
  - System reconstruction, personalization, customization
  - Human-agent interaction, interface management
Agent-based data mining infrastructure
Agent-driven data mining

- Agent service-based infrastructure
- Agentized trading rules and algorithms
- Agent ontology for rule/algorithm registration, in/out interface generation, etc
- Agentized rule/algorithm recommendation, subscription, reporting
- Message passing, request/response, dispatching among rules, interfaces, resources, reports, users
Control center
Financial Trading Rules Automated Development & Evaluation (F-TRADE 2.0)

F-TRADE Function Tree Configuration

Current Item ID: 3
Current Item Name: F-TRADE Function
Current Action: /managerAction.do
Action Type: showframe
Modify Item Delete Item

SubItem Name: Strategy Update
Action URL Object: /strategyUpdateAction.do
Action Type: update

Introduction

This is a web-based platform (F-TRADE) used to evaluate Stock Trading Signals and data mining algorithms. This platform is built on the huge historical and real stock information. It can be logged on from anywhere at anytime after registert.

Individual investors can choose any built-in trading algorithm (or strategy) for any stock they choose to evaluate the performance of your strategy in history. The users can try to set different values for the parameters to find out the best combinations for the best performance of the strategy.

Besides the above function, brokers can build your own trading algorithms to plug-in the platform online for your private use to evaluate the performance. You can benefit from the huge historical real stock information under the platform. You can also benefit from the testing to find the best combinations of the parameters.

The F-TRADE platform includes five centers which are User center, Algorithm center, Administration center, Control center, and Service center. For the new users, please click “User Manual” to read on-line user manual. For the experienced users, please click the relevant functional icons in the left column. For the inquiry, please send email to the Web Master.
Trading strategies of trading agents
Data mining-driven trading agents

- Data mining based trading rule agents
- KDD-driven trading agent optimizers with better rules and higher performance
- Mining actionable trading rules for trading agents in generic trading pattern set
- Parameter tuning of trading rule agents
- Trading rule recommenders
- Trading user assistants with better trading strategies
Pairs mining based trading agent
- Mining correlated stock pairs
- Correlated stock miner agent
- Stock pairs recommender
- Pairs trading strategy solution

![Diagram of data mining and trading strategy]

<table>
<thead>
<tr>
<th>Security1</th>
<th>Security2</th>
<th>StartDate</th>
<th>CloseDate</th>
<th>PriceDistance</th>
</tr>
</thead>
<tbody>
<tr>
<td>CBA</td>
<td>QMF</td>
<td>2002-1-1</td>
<td>2002-6-19</td>
<td>0.1</td>
</tr>
</tbody>
</table>

```
time_trade: 00:00:00
name_stocks: CBA:QMF
loss_c: 20:327044
standard: 32:27216
distance_l: 0.45:21568
```

![Graph of trading performance]
Conclusions

- Trading agent can support real-life smart trading
- Actionable trading strategies are essential
- Actionability enhancement, optimization, and integration are important
- Actionable trading support system are very useful
Acknowledgements

- Many people have contributed to this research:
  - From UTS:
    - A/Prof. Longbing Cao
    - Dr Jiarui Ni
    - Dr Li Lin
    - Dr Jiaqi Wang
  - From CMCRC
    - Prof. Michael Aitken
    - Prof. Alex Frino
The End

Thanks!

http://datamining.it.uts.edu.au
http://www.qcis.uts.edu.au