# Center for Robots and Sensors for the Human Well-being

ROSE-HUB

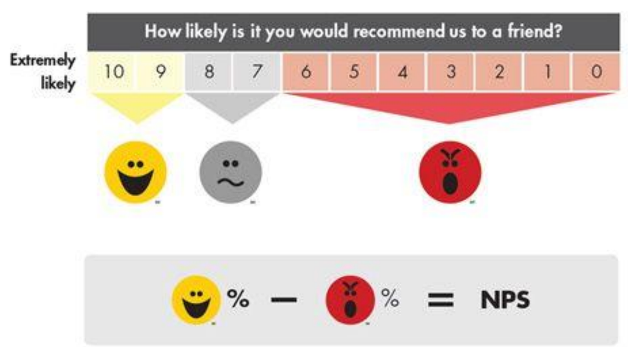
KDD Laboratory
College of IT, UNC-Charlotte

UNC CHARLOTTE

The Daniel Group

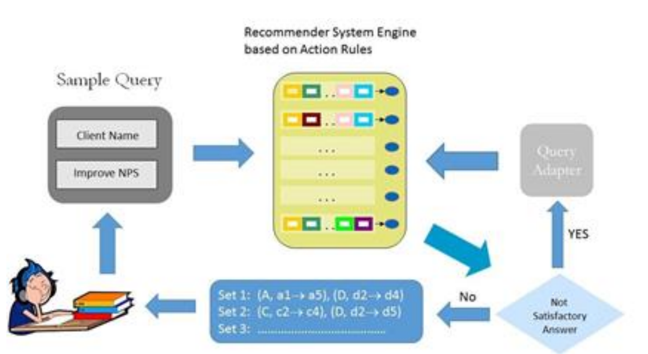
NSF

## Problem area – customer loyalty and satisfaction (Net Promoter Score)




**Net Promoter Score** (**NPS**) is a widely-used measure of business customer satisfaction and loyalty related to a product or service provider. Research It is built on a scale 1 to 10 where 1 means very unlikely to recommend the provider and 10 means highly likely to recommend. Based on observations of customers, referral and repurchase behaviors along such scale, customers' are divided into three logic levels: **promoter**, **passive** and **detractor**, which present customers' satisfaction, loyalty and likelihood of recommending this provider in a descending order.
**Promoters** are loyal enthusiasts who are buying from a company and recommend others to do so.
**Passives** are satisfied but unenthusiastic customers who are open to offers from competitors, while **detractors** are the least loyal customers who may urge others to avoid that company.
Customers are categorized into these three clusters based on their answers to the questions in questionnaires. Generally customers falling into interval 9-10 are seen as promoters, into 7-8 as passives, and into 0-6 as detractors. The partition into these three categories is widely accepted by business organizations. NPS is calculated by subtracting the percentage of customers who are detractors from the percentage of customers who are promoters.

## High-level architecture of the initial System




Examples of action rules:

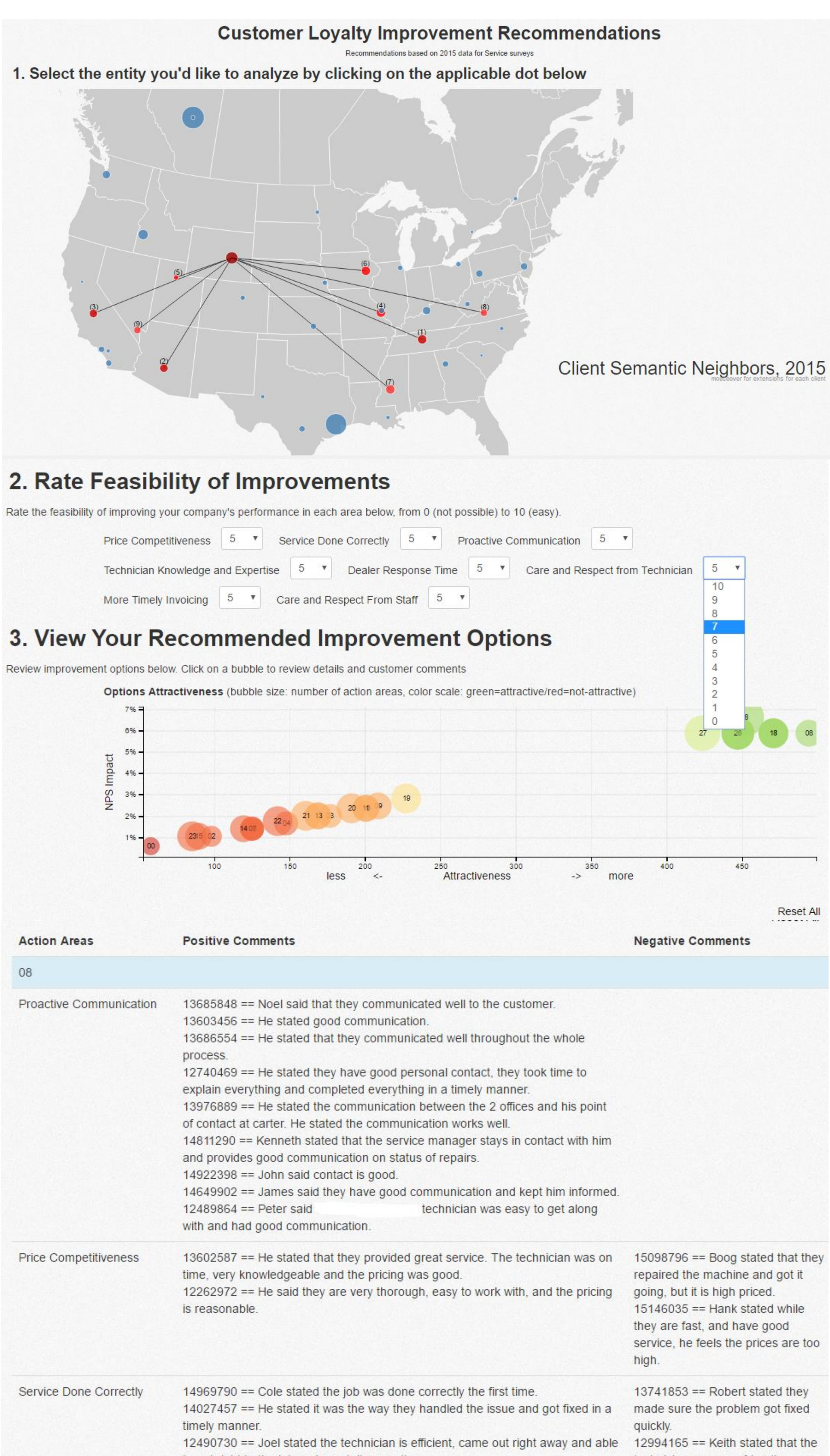
((Benchmark: All Overall Satisfaction, (1->10))*
(Benchmark: All Dealer Communication, (1->5)))
=>(Detractor->Promoter) sup= 5.0, conf= 100.0

((Benchmark: Service-Repair Completed When Promised, (8->3))*
(Benchmark: All Dealer Communication, (1->10))
=>(Detractor->Promoter) sup= 5.0, conf= 100.0

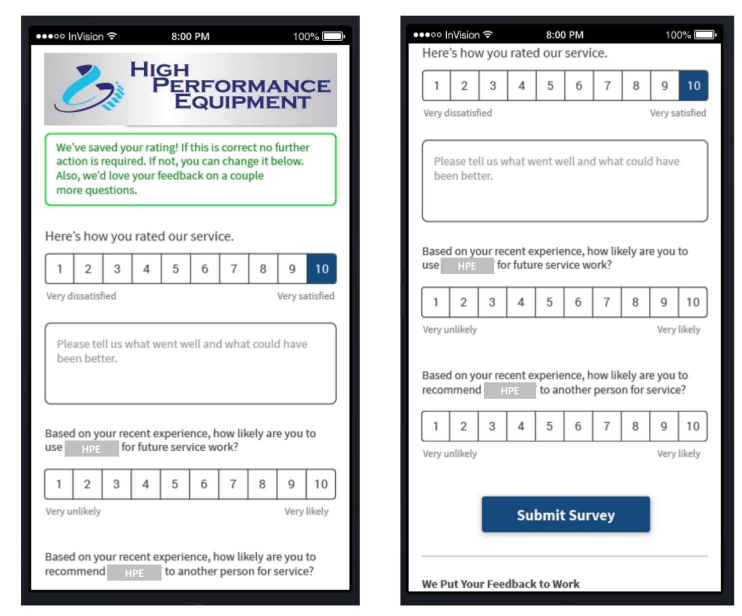
To realize the ultimate goal of adopting proper actions to improve the performance of every single client, in other words, improve its NPS rating with the given dataset, **Flexible Query Answering System** (**FQAS**) was initially designed to accept queries from a client regarding how to serve customers better and giving actionable suggestions to resolve such queries.
Hierarchically structured recommender systems are proposed - with leaves of the tree representing personalized recommender systems. Each personalized recommender system is responsible for providing valuable action rules for its corresponding client. To make the quality of retrieved action rules as high as possible, **Hierarchical Agglomerative Method for Improving NPS** (**HAMIS**) is proposed to maximally extend the dataset representing each client by using data of some neighboring clients having better NPS. It has been shown that the action rules generated from the extended dataset are more useful than from original dataset as they provide more options to clients, and have higher confidence.
The main procedures of FQAS: once a query from a client concerning the improvement of NPS ratings is submitted to FQAS, its corresponding recommender system will attempt to return action rules by following HAMIS. If action rules are not returned to clients, Query Adapter in FQAS would take over and a relaxed query is submitted to the recommender system. At the end, action rules with their triggers will be returned to targeted clients and tested for its effectiveness.

## Visualized Recommender System (demo)

Developed for 38 clients across all the United States and southern Canada, for different types of surveys




## Input: structured customer survey form

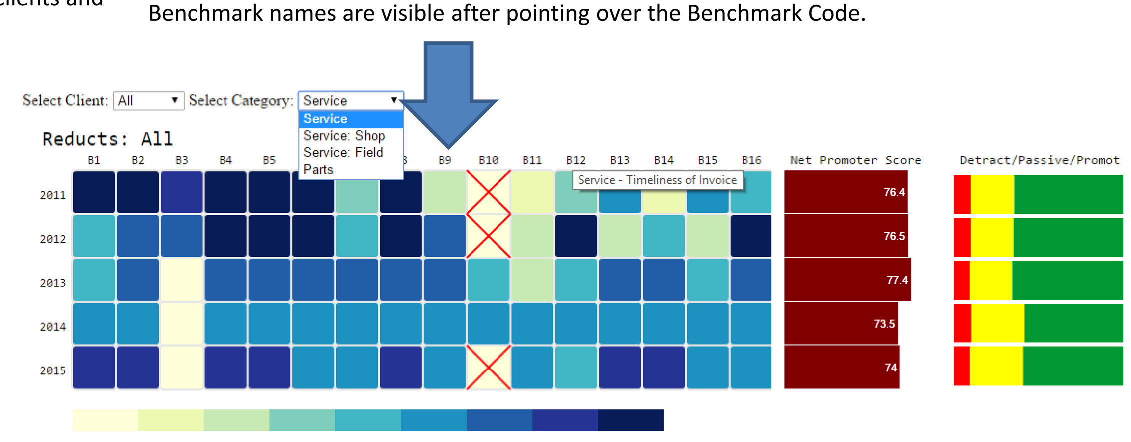



## Problem: Attribute analysis

The first problem we need to solve is to find out which benchmarks are the most relevant for Promoter Status. There is also a need to analyze how the importance of benchmarks changed over years for different clients (locally) and in general (globally), and additionally how these changes affected changes in Net Promoter Score, especially if this score deteriorated (which means customer satisfaction worsened). We need to identify what triggered the worst NPS drops and the highest NPS growths.

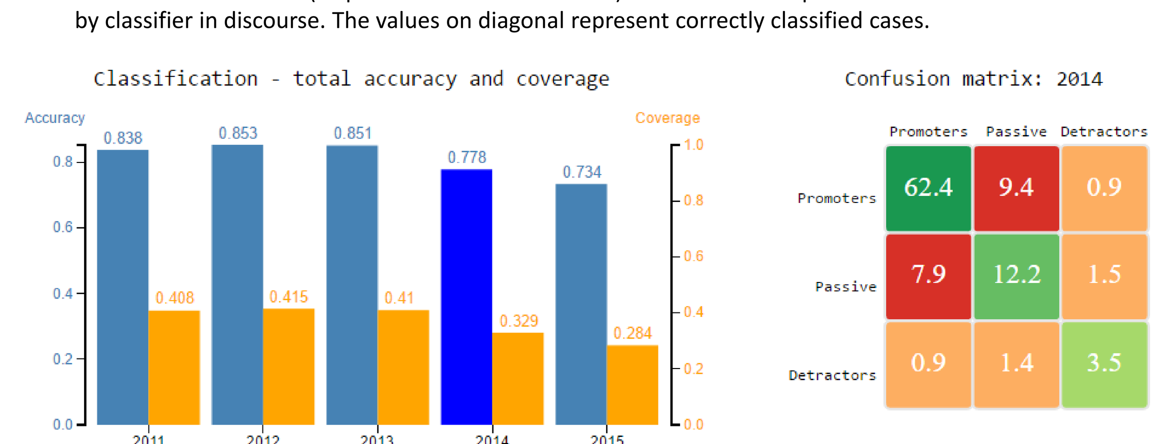
## Approach: Granular Computing (reducts) + Visualization

The **"Decision Reducts" matrix** visualizes minimal sets of benchmarks (reducts) that determine NPS (per chosen client's dataset in a given year).
Each colored cell in the matrix represents benchmark that was found in a reduct for the given year and client.
The color scale corresponds to occurrence percentage: darker cells indicate benchmarks that belong to more reducts than benchmarks represented by lighter cells. So, the darker the cell, the stronger the impact of the associated benchmark on promoter score. Red cross in a cell means that the benchmark has not been asked the client in the year. The average benchmark scores per year are visible after clicking on the client button and moving the mouse pointer over the cell. Benchmark names are visible after pointing over the Benchmark Code.




## Classification

To track the accuracy of the models built on the yearly client data we performed classification experiments on each client's dataset for each year. Evaluation was performed with 10fold cross validation on decomposition tree classifiers. We saved results from each classification task: accuracy, coverage, confusion matrix. Bar chart shows **accuracy** and **coverage** per yearly data. Details on classification accuracy for different categories are shown in a **confusion matrix** (updates after pointing over the accuracy/coverage bar for the corresponding year). Rows in this matrix correspond to actual decision classes (all possible values of decision) while columns represent decision values as returned by classifier in discourse. The values on diagonal represent correctly classified cases.



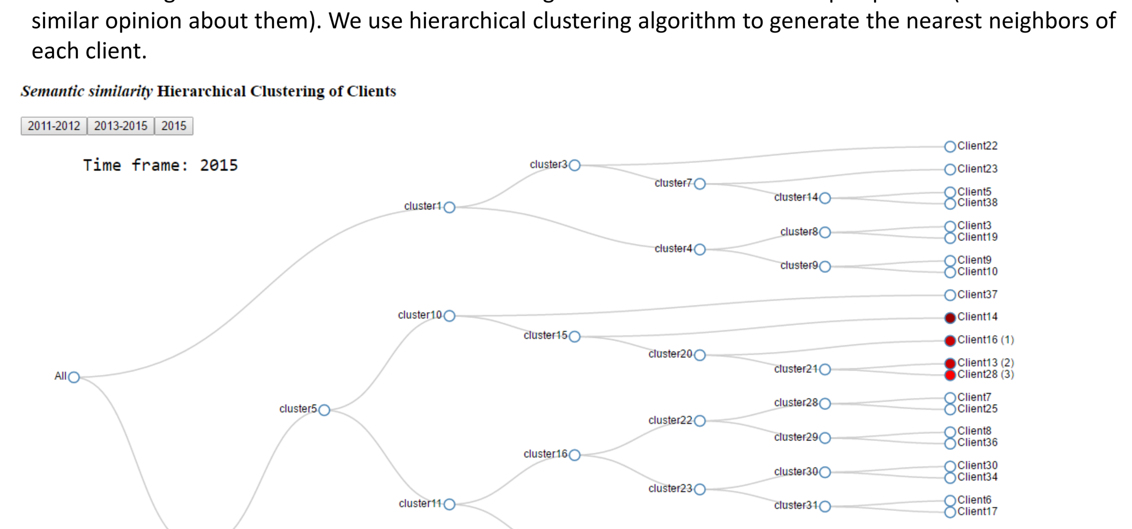

Confusion matrix: 2014

| | Promoters | Passive | Detractors |
|---|---|---|---|
| Promoters | 62.4 | 9.4 | 0.9 |
| Passive | 7.9 | 12.2 | 1.5 |
| Detractors | 0.9 | 1.4 | 3.5 |

## Clustering

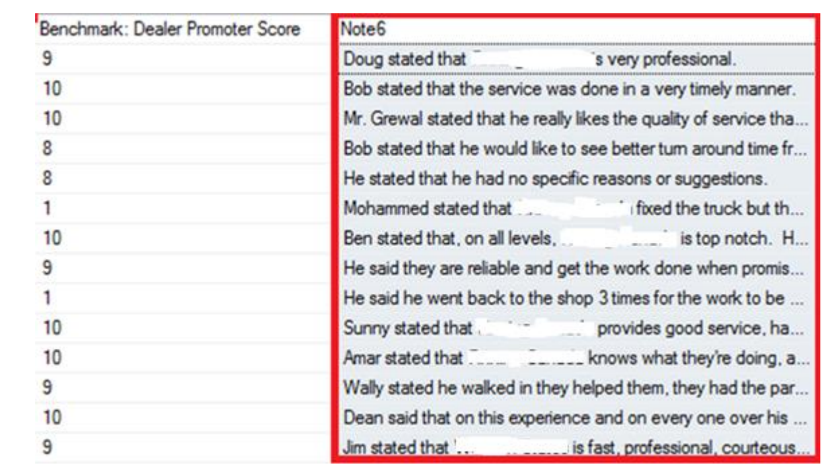
In order to recommend items (actions to improve in the service, products), we need to consider not only historical feedback of customers for this client, but we also propose looking at client who are similar in some way, but perform better. We use concept of semantic similarity to compare clients. Clients are compared in terms of the similarity of their knowledge concerning the meaning of three concepts: promoter, passive, and detractor. Clients who are semantically close to each other can have their datasets merged and the same considered as a single client from the business perspective (customers have similar opinion about them). We use hierarchical clustering algorithm to generate the nearest neighbors of each client.

*Semantic similarity* **Hierarchical Clustering of Clients**

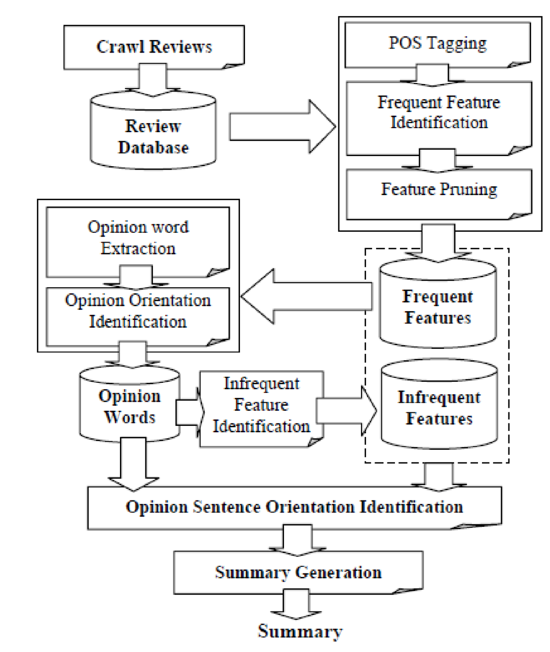



Clients form the domain for the agglomerative clustering algorithm based on their semantic distance. Clients are compared in terms of the similarity of their knowledge concerning the meaning of three concepts: promoter, passive, and detractor.
Clients which are semantically close to each other can have their datasets merged and the same considered as a single client from the business perspective (customers have similar opinion about them).
The dataset extensions are shown on dendogram (as colored client's nodes with corresponding number of extension).
The resulting dendrogram is a skeleton for the collection of hierarchically structured recommender systems. Lower the nodes in the dendrogram, more specialized the recommender systems are and the same the classifiers and action rules used to build their recommendation engines have higher precision and accuracy.
The recommendations are based on action rules which are extracted from the datasets assigned to all nodes of the dendrogram.
Higher a node in the dendrogram, the dataset assigned to it is larger - it is built by taking the union of all datasets assigned to the ancestors of that node.

## Input: unstructured text comments




## Natural Language Processing – Sentiment Analysis




## Domain-specific feature dictionary

| Feature Class | Subclasses | Examples of Seed Words |
|---|---|---|
| Service | completeness | correctly, properly, well, completed, fixed |
| | timeliness | timely, earlier, time(s), days, weeks, months |
| Communication | proactiveness | would like, respond, heard back |
| | ease of contact | difficult, hard, better, poor |
| | timeliness | timely, delay, quicker, slow, nobody |
| | quality (effectiveness) | effectively, failed |
| Staff | kindness | good, better, great, nice, friendly, gracious |
| | knowledgeability | knowledgeable, clueless, diagnosis, skill, professional, trained, inexperienced |
| | resource | enough, available, resourceful, staffed |
| Technician | kindness | good, better, great, nice, friendly, gracious |
| | knowledgeability | knowledgeable, clueless, diagnosis, skill, professional, trained, inexperienced |
| | resource | enough, available, resourceful, staffed |
| Invoice | accuracy | wrong, right, correct, incorrectly, correctly, incorrect, accurate, accurately |
| | expectation | match, matching, matched, expected |
| | timeliness | quick, quicker, quickly, slow, slowly, late, timely, time |
| Price | competitiveness | high, expensive, outrageous, fair, fairly, good, competitive, poor, excellent, reasonable, excessive |

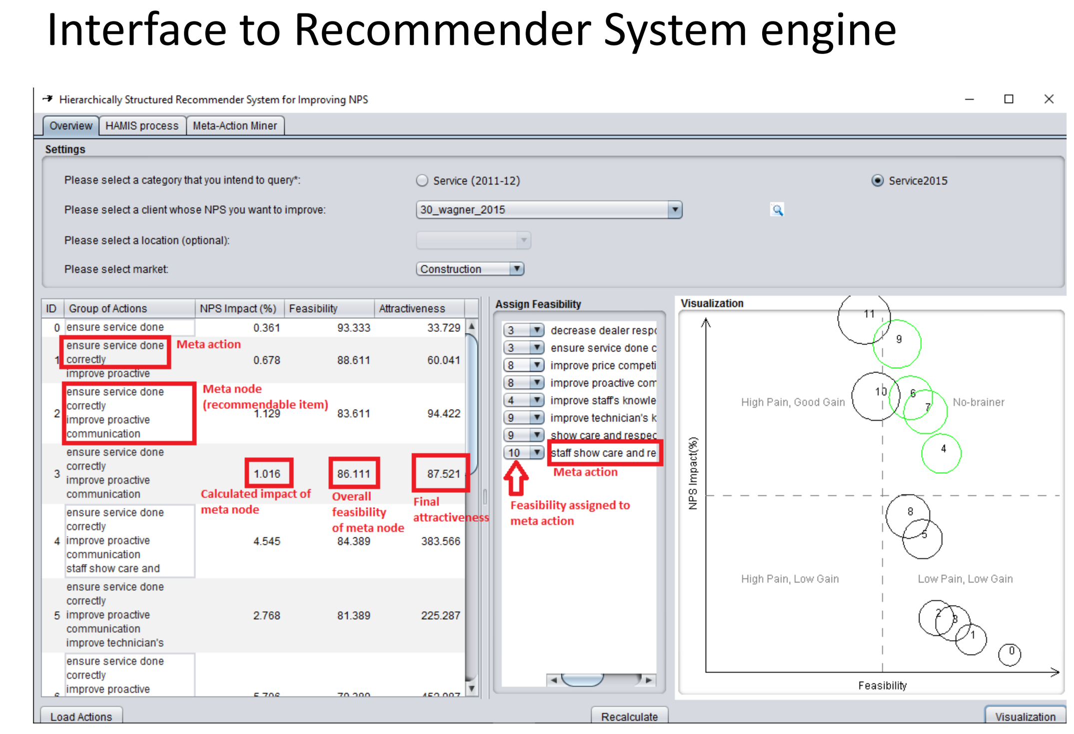
## Examples of Comments with Sentiment Orientation

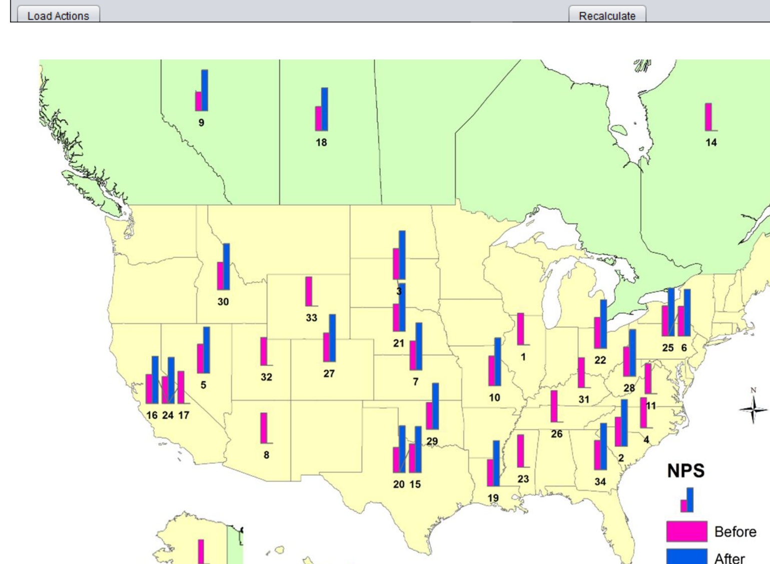
staff+++++,best manager=112
staff-----,bad experience because diagnosis=108
staff+++++,good technician=96
staff+++++,nice guy=87
staff+++++,excellent mechanic=85
staff+++++,great guy=83
staff+++++,excellent technician=79
staff+++++,good guy=74
staff+++++,wonderful dealer=71
staff+++++,good team=66
staff+++++,honest guy=60
staff+++++,pleased with manager=40
staff-----,not available technician=34
staff-----,wrong diagnosis=16
staff+++++,best mechanic=12
staff+++++,knowledgeable mechanic=6

invoice+++++,outstanding bill=141
invoice+++++,they billed properly=65
invoice+++++,fine invoice=61
invoice-----,refused pay bills=29
invoice+++++,happy with bill=12

price-----,not fair pricing=108
price+++++,good price=108
price+++++,fair pricing=87
price-----,aggressive pricing=66
price-----,unreasonable charge fee=37
price-----,not satisfied with price=35
price+++++,better pricing=27
price-----,expensive amount charged=12
price+++++,the charged fairly=12

## Interface to Recommender System engine




Columns in blue color show minimal expected improvement in NPS after following advice given by recommender system on clients level