A Novel Approach for Classifying Customer Complaints Through Graphs Similarities in Argumentative Dialogues

Boris A. Galitsky[a,*], María P. González[b,c,d], Carlos I. Chesñevar[b,c,*]

[a] *Birkbeck College, University of London*
Malet Street, London WC1E 7HX, United Kingdom
[b] *CONICET (National Council of Scientific and Technical Research),* Argentina
[c] *Department of Computer Science and Engineering, Universidad Nacional del Sur*
Alem 1253 – 8000 Bahía Blanca, Argentina
[d] *GRIHO CHI Research Group - Department of Computer Science – Universitat de Lleida*
C/Jaume II, 69 – 25001 Lleida, Spain

Abstract

Automating customer complaints processing is a major issue in the context of knowledge management technologies for most companies nowadays. Automated decision-support systems are important for complaint processing, integrating human experience in understanding complaints and the application of machine learning techniques. In this context, a major challenge in complaint processing involves assessing the validity of a customer complaint on the basis of the emerging dialogue between a customer and a company representative. This paper presents a novel approach for modelling and classifying complaint scenarios associated with customer-company dialogues. Such dialogues are formalized as labelled graphs, in which both company and customer interact through communicative actions, providing arguments that support their points. We show that such argumentation provides a complement to perform machine learning reasoning on communicative actions, improving the resulting classification accuracy.

Key words: automated decision making, automated complaint processing, argumentative dialogues, pattern matching,

1. Introduction and motivations

Customer complaint processing [9, 15] has become an important issue in the context of knowledge management technologies for large companies and organizations nowadays. Simply stated, complaint management can be seen as the

*Corresponding author
Email addresses: galitsky@dcs.bbk.ac.uk (Boris A. Galitsky), mpg@cs.uns.edu.ar (María P. González), cic@cs.uns.edu.ar (Carlos I. Chesñevar)

formal process of recording and resolving a customer complaint. Even though processing complaints is expensive (both in direct and indirect costs), companies can extract priceless knowledge from an appropriate complaint handling, with significant effects on customer retention rates and word-of-mouth recommendations (see e.g. [44, 10]). If complaints are transformed into knowledge about customers, they can provide a valuable business intelligence for enterprises. To exploit this intelligence, companies must design, build, operate and continuously upgrade systems for managing complaints, usually called *customer complaint management systems* (CCMS). In the last years several approaches have emerged to automatize CCMS, such as [26, 31, 1, 11], among others. Retailers and service providers may profit from such software services as they allow to handle complaints faster, providing the possibility of feedback analysis and datamining capabilities on the basis of a complaint database.

A typical complaint is a report of a failure of a product or service, followed by a narrative on the customer's attempts to resolve the issue. These complaints include both a description of the product or service failure and a description of the resulting interaction process (negotiation, conflict, etc.) between the customer and the company representatives. Since it is almost impossible to verify the actual occurrence of such failures, company representatives must judge the adequacy of a complaint on the basis of the *communicative actions* [15] provided by the customers in their narratives. Customers usually do their best to bring their points across, so that the consistency of communicative actions and the appropriateness of their arguments (represented as parameters of these actions) are major clues for the *validity* of their complaints. Indeed, a complaint narrative usually describes a conflict between an unsatisfied customer and customer support representatives, in which communicated claims need to be rationally justifiable by sound arguments. In contrast with the almost unlimited number of possible details regarding product failures, the emerging argumentative dialogues between customer and company can be subject to a systematic computational study. In this context, a major challenge in complaint processing involves distinguishing those customer complaints which are rationally acceptable from those which are not, so that the whole procedure of complaint handling can be better supported. Currently, most customer complaint management solutions are limited to the use of keyword processing to relate a complaint to a certain domain-specific class (e.g. ATM transactions via credit cards for banking complaints, as reported in [6]), or to the application of knowledge management techniques in software platforms for workflow processing (e.g. [31, 1, 8]). To the best of our knowledge, existing industrial CCMS platforms do not make use of natural language processing nor machine learning techniques to avoid slower performance, quality assurance and sustainability costs, so that most complaint handling functionalities are still manual. Thus, for example, even advanced tools such as Oracle PeopleSoft Enterprise Customer Relationship Management (CRM) [35] do not exploit possible benefits from related technologies belonging to the same company, such as Oracle Text for natural language processing [36]. In particular, no automated solutions have been developed to assess the validity of a customer complaint on the basis of the emerging dialogue between a customer and the company representatives, so that the whole procedure of complaint handling can be better supported.

In this paper we present a novel computational approach for modelling and classifying conflict scenarios associated with customer complaints. Complaint scenarios (representing complaint dialogues) will be labelled directed graphs, where nodes stand for the communicative actions associated with the conflict and labelled arcs denote conflict situations as well as the interaction flow between the two parties involved. Similarity matching among graphs will be applied to relate a particular scenario S to the class of *valid* or *invalid* complaint scenarios. We show how this approach can be embedded within *ComplaintEngine*, a software tool for automatic complaint processing. We provide experimental results showing that our proposal results in a better performance for automatically classifying complaints.

The rest of this article is organized as follows. First, in Section 2 we present the domain of complaint scenarios. We show how complaint scenarios can be modelled as labelled directed graphs, in which nodes correspond to communicative actions and directed arcs stand for temporal precedence and conflict relationships. Section 3 outlines the main components of *ComplaintEngine*, an automated software platform for processing customer complaints. We also discuss the original approach used in *ComplaintEngine* for classifying complaints as valid or invalid, based on analyzing sequences of communicative actions. Section 4 presents our proposal for classifying customer complaints through graph similarities in argumentative dialogues, extending the original conceptualization applied in *ComplaintEngine*. Section 5 discusses the three stages involved in the evaluation and assessment of our approach for solving real-world complaints extracted from a consumer advocacy website. Finally, Section 6 summarizes related work and Section 7 concludes.

2. Formalizing Customer Complaints through Complaint Scenarios

2.1. Understanding Customer Complaints

In what follows, we will focus on the domain of customer complaints which are submitted to public websites handling consumer advocacy issues. Such complaints include both a description of the product (or service) failure and a description of the resulting interaction process between a customer and company representatives. Usually, a complaint starts with a customer's belief that something went wrong with some product or service. The customer then contacts the company representatives with a request to replace (or fix, return, compensate, etc.) the product or clarify/explain the problem associated with this product. If the company's response is adequate in the customer's opinion, then no conflict is developed. Otherwise, the company may argue that the customer's claim is not valid, providing certain argumentation for this. The customer may insist that the complaint should be handled his/her way, presenting reasons for that, and the company may still disagree, requesting additional evidence for the customer's claims. There are several iterations in this process, after which the customer can finally decide that he/she will not deal with the company anymore (otherwise the complaint would be settled down), referring the complaint to a consumer advocacy enterprise.

In real life it is usually too expensive for companies (with respect to time and efforts for customer service representatives, required software infrastructure, etc.) to understand complaints thoroughly and to verify whether the claimed failure actually occurred. Indeed, it is frequently cheaper for a company to compensate for a product that is claimed to be faulty than to run an investigation. Therefore, companies may choose either to compensate all unhappy customers in one way or another, or just ignore their complaints. It is well known, however, that customers treat their complaints seriously, and adequate complaint handling is an important component of customer retention campaigns [44, 9].

Typically, customers use plain text to express their complaints. Analysis of textual complaints is a difficult task for natural language information retrieval because of several reasons, such as: a) a complex logical structure of a complaint; b) a number of interconnected inconsistencies; c) a biased representation of information in the uncertain conditions; d) a textual representation of a conflict with explicit and implicit goals; e) a rich diversity of technical and domain-specific terms; f) emotional and poorly organized structure; and g) a high number of ambiguities and unclear references. As a consequence, textual complaint information retrieval and understanding has not attracted much attention from the research community. Even though explicitly mentioned communicative actions can be extracted from text and processed reasonably well for achieving marketing intelligence [25], the treatment of implicit mental states and sentiments in text is still a significant challenge [34].

To overcome the bottleneck of natural language processing, complaint processing platforms (such as *ComplaintEngine*, presented in Section 3) provide specialized interactive forms to customers through which they can input their complaints. These forms can help to characterize relevant features in complaints, without loss of expressivity for the customer. Such relevant features are mainly associated with the communicative actions provided by customer and company in a complaint narrative, as well as their interrelationships. In the next Section we will present the notion of *complaint scenario*, a graph-based characterization for capturing the most relevant features of customer-company dialogues.

2.2. Complaint Scenarios: modelling Customer Complaints using Graphs

As discussed before, complaints are usually presented in natural language. When writing a complaint letter, it is expectable that customers may become very emotional and passionate, basing their argumentation on feelings rather than on logic. This brings in disadvantages both on the customer and the company side, as it is harder for a company to evaluate the complaint validity, and the customer may not be able to bring his/her point across following a rational argumentation line.

In order to provide a more formal approach to represent customer complaints, we will define the notion of *complaint scenario*, a graph-based formalization for representing customer-company dialogues. In such scenarios we will distinguish a number of *communicative actions* [15], which from empirical evidence have proven to be representative for characterizing different possible interactions between customer and company in a complaint scenario (see discussion in Section 4.1). Such actions will correspond to vertices in a graph, connected by means of *temporal* and *attack relationships*. Temporal relationships formalize the order in which actions were advanced in a complaint dialogue, whereas attack relationships help to identify conflicting situations. Next we will formalize these concepts.

Definition 1 (Communicative action). *A* communicative action *is a functor of the form* verb(agent, subject, cause) *where* verb *characterizes some kind of interaction between customer and company in a complaint scenario (e.g.,* explain*,* confirm*,* remind*,* disagree*,* deny*, etc.),* agent *identifies either the customer or the company,* subject *refers to the information transmitted or object described, and* cause *refers to the motivation or explanation for the subject.*

Thus, for example, a communicative action associated with some customer claim such as *"I disagreed with the overdraft fee you charged me because I made a bank deposit well in advance"* would be represented as

disagree(customer, "overdraft fee", "I made a bank deposit well in advance").

Scenarios are intentionally simplified as labelled directed graphs to allow for effective similarity matching among them. Each vertex in the graph will correspond to a communicative action. An arc (oriented edge) may denote either *temporal precedence* or an *attack relationship* between two actions a_i and a_j. In the first case, we will distinguish between consecutive actions which refer to the *same subject* from those which refer to *different subjects*. Graphically, we will distinguish these situations by means of **thick arcs** and ***thin arcs***, respectively. Attack relationships, on the other hand, indicate a conflict between two communicative actions. Formally:

Definition 2 (Complaint scenario). *A* complaint scenario *is a labelled directed graph* $G = (V, A)$*, where* $V = \{action_1, action_2, \ldots, action_k\}$ *is a finite set of vertices corresponding to* communicative actions*, and* $A = A_{thick} \cup A_{thin} \cup A_{attack}$ *is a finite set of labelled arcs (ordered pairs of vertices), classified as follows :*

- *Each arc* $(action_i, action_j) \in A_{thick}$ *corresponds to a* temporal precedence *of two actions* (v_i, ag_i, s_i, c_i) *and* (v_j, ag_j, s_j, c_j) *referring to the* same *subject (that is,* $s_i = s_j$*).*
- *Each arc* $(action_i, action_j) \in A_{thin}$ *corresponds to a* temporal precedence *of two actions* (v_i, ag_i, s_i, c_i) *and* (v_j, ag_j, s_j, c_j) *referring to a* different *subject (that is,* $s_i \neq s_j$*).*
- *Each arc* $(action_i, action_j) \in A_{attack}$ *corresponds to an* attack relationship *between* $action_i$ *and* $action_j$*, indicating that the cause of* $action_i$ *is in conflict with the subject or cause of* $action_j$*.*

Graphs associated with complaint scenarios have some distinguished features: 1) All vertices are ordered in time, so that there is one incoming arc and one outgoing arc for all vertices (except the initial one and terminal one); 2) For thick and thin arcs, at most one incoming and only one outgoing arc are admissible; 3) For attack arcs, there can be many outgoing arcs from a given vertex as well as many incoming arcs. The vertices involved may be associated with different agents (e.g. customer and company) or with the same agent (i.e. when the customer contradicts himself). To compute similarities between graphs, sub-graphs of the same configuration with similar labels of arcs and strict correspondence of vertices will be analyzed.

Example 1. *Consider the text in Fig. 1(a) representing a complaint scenario in which a client is presenting a complaint against a company because he was charged with an overdraft fee unfairly (according to the customer's viewpoint). We denote the parties in this complaint scenario as* ***Pro*** *and* ***Con*** *(proponent and opponent, or equivalently customer and company), to stress the fact that we are in a dialectical setting. In this text communicative actions are shown in boldface, and some expressions within the text appear underlined, indicating that they are the focus of the attacks to earlier statements. Fig. 1(b) shows the associated graph, where straight thick and thin arcs represent temporal sequences, and curve arcs denote attack relationships. As explained before, note that edges in the graph in Fig. 1(b) are thick when they refer to actions whose subjects stay the same and thin when they change.*

According to Def. 2, the situation from Fig. 1(b) is formalized as a graph $G = (V, A)$*, where* $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$*, with:*

v_1 = *explain(customer, "I made a deposit", "I wrote a check")*
v_2 = *confirm(company, "the deposit is not available", "it takes a day to process the deposit")*
v_3 = *remind(customer, "overdraft unfairly charged", "the same happened one month ago")*
v_4 = *explain(company, "overdraft fee due to insufficient funds", "disclosed from account information")*
v_5 = *disagree(customer, "overdraft fee", "I made the bank deposit well in advance")*
v_6 = *deny(company, "responsibility", "nothing can be done at this point")*

and $A = A_{thick} \cup A_{thin} \cup A_{attack}$*, with* $A_{thick} = \{(v_1, v_2), (v_3, v_4), (v_4, v_5)\}$*,* $A_{thin} = \{(v_2, v_3), (v_5, v_6)\}$ *and* $A_{attack} = \{(v_4, v_1), (v_5, v_2)\}$*.*

Note the correspondence between the first part of the complaint dialogue and the graph: the same thing that was confirmed *had been previously* explained *(thick edge), and another (different) thing was later on* reminded *(thin edge). Note that first two sentences (and the respective subgraph comprising two vertices) are about the current transaction (*deposit*), three sentences after (and the respective subgraph comprising three vertices) the customer addresses the* unfair charge*, and the customer's last statement is probably related to both issues above. Hence the vertices of two respective subgraphs are linked with thick arcs:* explain-confirm *and* remind-explain-disagree*. It must be remarked that the underlined expressions help to identify where conflicts in the dialogue arise. Thus, the company's claim as* disclosed in my account information *attacks the*

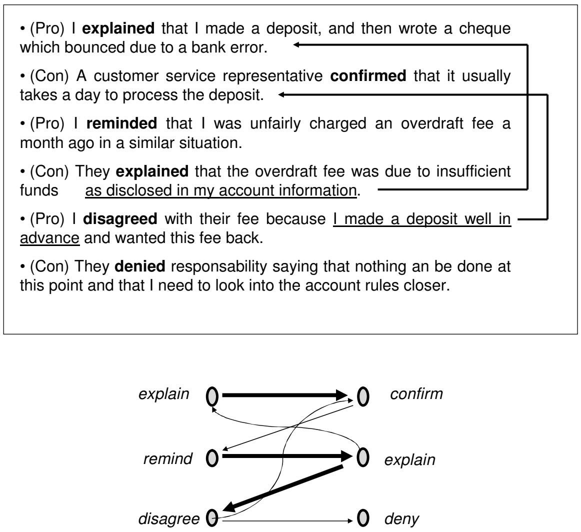

Figure 1: a) A complaint scenario with attack relations (top); b) associated graph representation (bottom)

customer's assertion due to a bank error. *Similarly, the expression* I made a deposit well in advance *attacks the statement* it usually takes a day to process the deposit *(makes it non-applicable). The former attack has the intuitive meaning* "existence of a rule or criterion of procedure attacks an associated claim of an error"*, whereas the latter would have the meaning* "the rule of procedure is not applicable to this particular case"*.*

As discussed in the introduction, a major challenge in complaint processing involves assessing the validity a customer complaint on the basis of the emerging dialogue between a customer and a company representative. Complaint advocacy services rely on human experts for classifying complaints as the one described in Fig. 1, distinguishing those which are *valid* (in a logical sense) from those which are not. *Valid* complaint scenarios are those in which the customer follows a sound argumentation line as the dialogue with the company proceeds. On the contrary, *invalid* complaint scenarios are those which contain some kind of ill-formed reasoning (fallacies) in the argumentation process. Several fallacies are possible (e.g., the customer performs circular reasoning, coming back to something that was already explained; or the customer contradicts himself, attacking some statement which he had previously granted as accepted).[1]

Formalizing complaints in terms of complaint scenarios can be helpful for automatically detecting suspicious complaint dialogues on the basis of the communicative actions involved. Indeed, complaint scenarios can help to identify

[1]For an in-depth discussion of the role of fallacies in argumentation, see [27, 41].

subtle aspects in such dialogues which make them ill-formed or fallacious. Thus, the complaint scenario in Example 1 seems ill-formed, as apparently the complainant does not understand the procedure of *processing the deposit* nor distinguishes it from an *insufficient funds* situation. Note that the scenario itself does not have surface-level explicit inconsistencies. At the first sight, the complainant's plot in terms of communicative actions seems normal, and the complainant's arguments sound reasonable. Nevertheless, by looking deeper into the case (without taking into account banking knowledge), a problem becomes visible. Rather then accepting the opponent's *confirmation* about subject S, the complainant switches from S to another subject S' (*reminds* about another transaction), and *disagrees* with the opponent's explanation of this new subject, mixing both of them. Moreover, from the complainant's perspective, his opponent reacts with a *denial* to his *disagreement*. In other words, the complainant disagrees with what has already been explained but at the same time "attacks" what has granted as confirmed, which is a suspicious argumentation pattern.

3. The *ComplaintEngine* platform

The *ComplaintEngine* platform [14, 21][2] was designed to help customers and companies during the complaint process, helping customers in the process of writing a sound complaint. The facilities provided by the *ComplaintEngine* platform aim at reducing decision regret. Regret is a post-decision feeling regarding not having chosen a better alternative (compensating for an invalid complaint, or ignoring valid complaint). Recent behavioral research [29] has indicated that, in addition to pursuing higher performance and user satisfaction, reducing decision regret is another important consideration for many decision-makers.

The graph-based notion of complaint scenario presented in the previous Section provided the theoretical background for the design of the *ComplaintEngine* platform. Two major facilities for complaint handling are provided, namely: a) an *interactive complaint form*, which helps the customer to express his/her complaint through a suitable template which can be directly translated into a complaint scenario ; and b) a *validity check button*, which allows the customer to evaluate whether his/her complaint is not fallacious or ill-formed.

In *ComplaintEngine*, the interactive complaint form provides a template for introducing the elements characterizing a complaint scenario (communicative actions and temporal and attack relationships among them). The form includes a pull-down menu with a list of pre-selected communicative actions to select from, and text fields to specify the action parameters. The user interface to specify a complaint scenario in the *ComplaintEngine* is shown in Fig. 2. A complainant (e.g., a customer) selects his communicative actions (on the left)

[2]For space reasons, we restrict ourselves to a description of the main elements in the *ComplaintEngine* related to our proposal. For more details see [14, 21].

and communicative actions of his opponent (e.g., a company, on the right) respectively. Communicative actions are selected from a list of twenty or more, depending on the industry sector of the complaint. The parameters of communicative actions are specified as text in the interactive form (even though there is no underlying graph-based complaint scenario representation). Communicative actions selected by the user in the list boxes constitute the vertices of such a graph, whereas check boxes on the right of the list boxes are used to specify whether the incoming arc is **thick** (**checked**) or *thin* (*unchecked*). Finally, check boxes linked with a vertical line (see Fig. 2) are used to specify *attack links* between different communicative actions.[3]

On the basis of the customer input, the *ComplaintEngine* is able to provide immediate feedback concerning the validity status of the complaint (*justified* or *unjustified*, indicating whether the complaint proceeds or not. This is done by contrasting the current complaint with other previous complaints already solved, stored in a database. This validity status is computed by applying pattern matching on the sequence of communicative actions used in the form with respect to those previous complaints in the database. As an additional justification for the customer (user), the *ComplaintEngine* backs up its decision by highlighting the cases which are similar to the one to be classified, and which are different from it [21, 14]. After a complaint is partially or fully specified, the user can evaluate its consistency using the validity status button. *Complaint-Engine* may issue a warning or advice concerning improvement of the logical structure of this complaint. When the complainant is finally satisfied with the validity status of his/her complaint form, he/she can submit the completed form to the company, using electronic submission facilities also provided by the *ComplaintEngine* suite. It must be remarked that a complainant has the choice to use the above form or to input complaint as a text, and a specialized linguistic tool processes that text and fills in the form for him/her. However, using the form as a "template" encourages complainants to enforce a logical structure on their complaints. Moreover, in contrast to communicative actions, it is too hard for current automated text-processing technology to reveal attack relationships from text. In that respect the template proves to be particularly useful, as attack links can only be defined via the form using arrows.

The *ComplaintEngine* was originally designed as a traditional NLP application, where statistical keyword analysis was applied to solve the problem of complaint classification. In that first version, complaint scenarios were graphs in which no attacks were considered. Vertices were connected only by temporal relationships (no distinction between thick and thin arcs was required), so that the whole graph could be conceptualized just as a sequence of communicative actions [14, 21]. In order to analyze the role of attacks relationships for classifying complaints we developed an alternative approach, based on applying

[3] In the *ComplaintEngine*, the representation of complaints via the interactive complaint form assumes that there is a single communicative action per step. These actions are shown in groups of three for the sake of visualization.

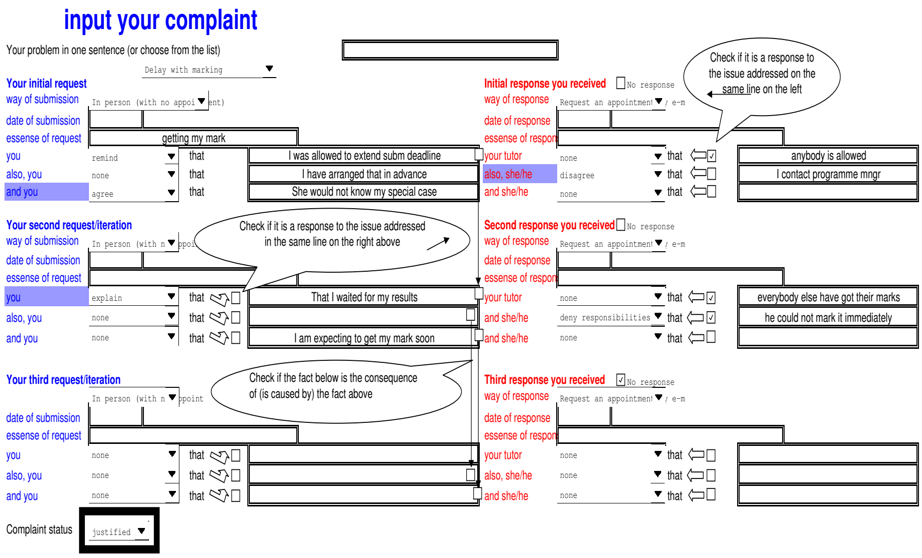

Figure 2: Current user interface for the Interactive Complaint Form in the *ComplaintEngine*

supervised learning through graph similarities in complaint scenarios. As we will see in the next Section, our proposal involves the integration of a number of criteria for finding similarities between communicative actions on the basis of their attributes. New complaint scenarios will be classified using supervised learning, based on the training dataset of previous complaints scenarios which have been already analyzed by human experts and tagged as valid or invalid.

4. Classifying Complaint Scenarios through Graph Similarities

In order to classify customer complaints according to their validity, the original *ComplaintEngine* suite [14, 21] considered complaint scenarios as just sequences of communicative actions, without taking attack relationships among such actions into account. Such sequences were indeed connected graphs, involving only temporal precedence relationships between communicative actions. Our alternative formalization of complaint scenarios presented in Section 2 extends that notion by including attack relationships, which properly identify the conflicts in the argumentative dialogue between customer and company representative.

In order to assess the status of a new customer complaint C, we will analyze the structure of the labelled graph G_C (characterizing the complaint scenario associated with C) to determine if it can be associated with the class of valid complaint scenarios. This classification will be performed using a similarity criterion which determines whether G_C can be related to some previous graph in the dataset which was tagged as valid. In the next subsections we will formalize the similarity criterion used for comparing the graph G_C with those graphs in the training dataset. We will show how communicative actions involved in any particular complaint scenario can be clustered into different categories, on the basis of their inherent attributes and other elements from speech act theory [39, 2]. This clustering will help to define similarity among graphs corresponding to complaint scenarios by means of maximal common subscenarios.

4.1. Modelling communicative actions and their attributes

Speech act theory [39, 2] is one of the most promising approaches to categorizing communicative actions in terms of their roles. In general, speech acts (also called illocutionary acts)[4] are acts of communication which express attitudes; the type of speech act being performed corresponds to the type of attitude being expressed. Thus, for example, a statement expresses a belief, a request expresses a desire, and an apology expresses a regret. As an act of communication, a speech act succeeds if the audience identifies, in accordance with the speaker's intention, the attitude being expressed [3]. As pointed out in [4], attitudes can be seen as relational mental states connecting a person to a proposition, expressing meanings or content that can be true or false. Thus, a person can have different mental postures towards a proposition, such as believing, desiring, or hoping.

Following [4], four major categories can be identified for classifying illocutionary speech acts: *stating*, *requesting*, *promising* and *apologizing*. Although speech act theory relates every speech act to a single category, for our purposes speech acts will be allowed to belong to multiple categories. The reason for this is that we are modelling a two-party scenario, in which the beliefs, desires and intentions of each party differ. Thus, for example, the action *confirm* could belong to the *stating* category (e.g., the customer confirms he paid in advance) as well as to the *apologizing* category (e.g., the company confirms that the customer's payment will be reimbursed).

In order to define a robust framework to find similarities between complaint scenarios, we are interested in distinguishing which are the most common communicative actions used in complaint scenarios, and how they can be clustered in terms of the attitudes commonly associated with them. In order to do this, we identified the set S_{freq} of those communicative actions which are most frequently used in conflict situations (see Fig. 4(a)). This was done empirically by collecting statistically significant occurrences of verbs for communicative actions in complaint texts [15]. Such communicative actions were made available as possible options for entering a complaint in the Interactive Complaint Form of the *ComplaintEngine* (Fig. 2). Since in our context every communicative action could belong to more than one speech act category, we identified five different attributes (see Fig. 3) associated with every communicative action [13]. These attributes are related to additional semantic information contained in the communicative action (friendliness, reactivity, informativeness, confidence,

[4]The term "speech act" is often meant to refer just to the same thing as the term "illocutionary act", originally introduced by John L. Austin [2].

- **Friendliness (FR)**: expresses whether a communicative action is a cooperative (friendly, helpful) move (1), uncooperative (unfriendly, unhelpful) move (-1), or hard to tell (0).
- **Reactivity (RE)**: specifies whether a communicative action is expected to be followed by a reaction (1), constitutes a response which follows a previous request (-1), or hard to tell (0).
- **Informativeness (IN)** tells if a communicative action brings in additional data about the conflict (1), does not bring any information (-1).
- **Confidence (CO)** specifies the confidence associated with choosing a particular communicative action: high knowledge/confidence (1), or lack of knowledge/confidence (-1).
- **Emotion (EM)** tells about the potential emotional load of the participant: high (1) or low (-1).

Figure 3: Possible values for attributes in communicative actions

emotion) represented by different numerical values.[5] On the basis of these attributes and their values, we were able to characterize every communicative action in S_{freq} [15], as shown in Fig. 4(b).[6] Thus, for example, the communicative action *agree* involves a positive attitude from the speaker (1), a response to a previous request (-1), no information content (-1), high confidence in choosing this particular action (1), and low emotional load (-1).

Formal concept analysis [23] was used to characterize the set S_{freq} of communicative actions in the context of our framework. In formal concept analysis, a (formal) *context* consists of a set of objects O, a set of attributes A, and an indication of which objects have which attributes. A *concept* is a pair containing both a natural property cluster and its corresponding natural object cluster. A "natural" object cluster is the set of all objects that share a common subset of properties, and a "natural" property cluster is the set of all properties shared by one of the natural object clusters. Given a set of objects O and a set of attributes A, a *concept* is defined to be a pair (O_i, A_i) such that 1) $O_i \subseteq O$; 2) $A_i \subseteq A$; 3) every object in O_i has every attribute in A_i; 4) for every object in O that is not in O_i, there is an attribute in A_i that the object does not have; 5) for every attribute in A that is not in A_i, there is an object in O_i that does not have that attribute. Given a concept (O_i, A_i), the set O_i is called the *extent* of the concept, and the set A_i is called the *intent*. Concepts can be partially

[5] An in-depth analysis of the criteria used for defining attributes for communicative actions as well as their associated numerical values is outside the scope of this article, and has been addressed elsewhere [15, 17].

[6] The attribute values associated with every communicative action were adopted on the basis of empirical evidence when processing customer complaints [17].

ordered by inclusion: if (O_i, A_i) and (O_j, A_j) are concepts, a partial order $\leq$ can be defined, where $(O_i, A_i) \leq (O_j, A_j)$ whenever $O_i \subseteq O_j$. Equivalently, $(O_i, A_i) \leq (O_j, A_j)$ whenever $A_j \subseteq A_i$. In general, attributes may allow multiple values (many-valued attributes), characterizing many-valued contexts. By applying so-called *conceptual scaling*, many-valued contexts can be transformed to one-valued scaled contexts from which concepts can be computed. The family of these concepts obeys the mathematical axioms defining a lattice, and is called a *concept lattice* or *Galois lattice*.[7] So called *line diagrams* [43] are used in order to succinctly represent information about intents and extents of formal context in a concept lattice. Nodes are circles that can be labelled with a) both attributes and objects; b) attributes; c) objects or d) none. In order to consider some distinguished labels, some nodes appear as circles which are half-filled in their lower part (labelled with objects only), and nodes which are half-filled in their upper part (labelled with attributes only). Nodes which are empty circles have no particular labels.

In order to provide a formal characterization of the communicative actions in S_{freq} in terms of their attributes a concept lattice was obtained. Nominal scaling was applied on the first and second attributes (the third, fourth and fifth attributes were already two-valued). [8] As a result of this scaling, we obtained nine two-valued attributes associated with different possible values of the original five attributes: PosAtt (FR=1), NegAtt (FR=-1), Request (RE=1), Respond (RE=-1), InfoIn (IN=1), High_Conf (CO=1), Low_Conf (CO=-1), Intense (EM=1), Relaxed (EM=-1). It must be remarked that some particular two-valued attributes derived from the original attributes (namely those corresponding to FR=0, RE=0, and IN=-1) are not considered for building the resulting concept lattice shown in Fig. 5(top), as they do not contribute strongly in distinguishing communicative actions from each other. The resulting scaled context had nine two-valued attributes, resulting in the concept lattice shown in Fig. 5(top).

The ConExp software [43] was used to construct and visualize the concept lattice of communicative actions and their associated nine two-valued attributes. Some selected nodes are provided with descriptions of the corresponding "intents" and "extents" [23] subscribed to show how certain communicative actions are semantically related to each other. The concept lattice illustrates the semantics of communicative actions, and shows how to cover different meanings in the knowledge domain of customer-company interaction in complaint scenarios. The clustered view shown in Fig. 5 (bottom) is an alternative way to visualize the semantic model of communicative actions, complementing the information provided by the concept lattice. It is used to verify that the attributes of communicative actions have been selected properly (i.e. that the space of meanings

[7] An in-depth discussion of the underlying definitions and properties of formal concept analysis is outside the scope of this article. For more details see [23].

[8] Nominal scaling involves transforming multi-valued attributes into two-valued attributes. Thus, for example, from the FR attribute with values $\{-1, 0, 1\}$ we get three two-valued attributes PossAtt (yes/no), NegAtt (yes/no), NeutralAtt (yes/no).

Customer describes action about himself	Customer describes actions about the company
Agree, explain, suggest, bring company 's attention, remind, allow, try, request, understand, inform, confirm, ask, check, ignore, convince, disagree, appeal, deny, threaten	Agree, explain, suggest, remind, allow, try, request, understand, inform, confirm, ask, check, ignore, convince, disagree, appeal, deny, threaten, bring to customer's attention, accept complaint, accept/deny responsabilities, do/do not understand problem, encourage, cheat

Communicative Action	Attributes				
	FR	RE	IN	CO	EM
agree	1	-1	-1	1	-1
accept	1	-1	-1	1	1
explain	0	-1	1	1	-1
suggest	1	0	1	-1	-1
bring_attent	1	1	1	1	1
remind	-1	0	1	1	1
allow	1	-1	-1	-1	-1
try	1	0	-1	-1	-1
request	0	1	-1	1	1
understand	0	-1	-1	1	-1
inform	0	0	1	1	-1
confirm	0	-1	1	1	1
ask	0	1	-1	-1	-1
check	-1	1	-1	-1	1
ignore	-1	-1	-1	-1	1
convince	0	1	1	1	-1
disagree	-1	-1	-1	1	-1
appeal	-1	1	1	1	1
deny	-1	-1	-1	1	1
threaten	-1	1	-1	1	1

Figure 4: a) Set S_{freq} of communicative actions most frequently used by customers in their complaints (on the left); b) associated attribute values for elements in S_{freq} (on the right).

for communicative actions is covered evenly).

Similarities between communicative actions allow to cluster them according to their attribute values. Assuming that $a_1 = (x_1, x_2, x_3, x_4, x_5)$ and $a_2 = (y_1, y_2, y_3, y_4, y_5)$ are the attribute values associated with two particular communicative actions, similarities vertices are represented as 5-tuples (s_1 s_2 s_3 s_4 s_5) whose values are computed on the basis of the attributes values in a_1 and a_2. We compute the similarity for every s_i as follows: $s_i = 1$ iff $x_i = 1$ and $y_i = 1$; $s_i = 0$ iff $x_i = 0$ and $y_i = 0$; otherwise s_i is assigned to "x". The graph in Fig. 5 (bottom) displays the similarities between communicative actions expressed through their attributes. Only close similarities are shown: deviation by one attribute (solid box) and by two attributes (dashed box). It must be noted that in the graph in Fig. 5 (bottom) we can distinguish two main "clusters":

- The cluster of communicative actions associated with negative attitudes which *do not supply information* (on the right bottom). These communicative actions are similar to each other, deviating from *deny* by one attribute out of five. Also, the difference between *deny*, *appeal* and *threaten* is the second attribute only, and therefore their similarity is expressed by the same vertex (-1 x -1 1 1). Moreover, three similarity vertices for this cluster converge to the similarity vertex for the whole cluster (-1 x -1 x x), highlighted by an ellipse.

- The cluster corresponding to the rest of communicative actions, which are connected with each other and linked with the above cluster by the *deny/accept* link. Communicative actions of this cluster are not as "dense" as ones for the other cluster, as most of them are different from each other by two attributes out of five.

4.2. Capturing Similarities between Complaint Scenarios

As stated before, our ultimate aim is to classify a new complaint scenario C (characterized by its underlying graph representation) as valid or invalid, according to the communicative actions used by the customer and the attack relationships established during the customer-company dialogue. Valid complaints will be those which are rationally sound (i.e, there is a consistent plot along the customer-company dialogue), whereas invalid complaints will be those which are fallacious or inconsistent. Our approach will be based on applying supervised learning on a training dataset of complaint scenarios which have been already classified as valid or invalid by human experts. Samples of these two kinds of complaint scenarios are depicted in Fig. 6(left) and Fig. 6(right), respectively. In every of such scenarios the vertices on the left side (resp. right side) denote actions of the customer (resp. company). According to the convention already introduced in Def. 2, thin and thick arcs link vertices indicating a temporal precedence (earlier-later), whereas curly arcs correspond to attack relationships.

It must be noted that every complaint scenario in the training dataset includes sequences of communicative actions which alternate the first attribute (customer/company) while referring to the same subject. Such sequences are called *interaction steps*, and are in fact distinguished paths in the graph associated with a complaint scenario, involving between two and six vertices.[9] Graphically, communicative actions in an interaction step will be connected by thick arcs, following the convention introduced in Section 2. Thus, for example, *suggest* in scenario V_2 (Fig. 6) is linked by a thin arc to the communicative action *remind*, making clear that the subject of the suggestion is not related to the subject of what is being reminded by the company. In V_2 the interaction step *remind-accept-ignore-threaten* can be identified, as these communicative actions refer to the same subject (and consequently are denoted in V_2 as vertices linked by thick arcs).

Interaction steps will help to identify similarity between complaint scenarios in terms of maximal common subscenarios. Our approach will be based on the methodology for integrating formal concept analysis and version spaces presented in [22], which allow us to characterize the notion of *similarity* between two graphs representing complaint scenarios.[10] Given two complaint scenarios $C_X = (V_X, A_X)$ and $C_Y = (V_Y, A_Y)$, the similarity between C_X and C_Y (denoted $C_X * C_Y$) is defined as the set $\{G_1, G_2, \ldots, G_k\}$ of all inclusion-maximal

[9] Experience shows that typical complaints submitted to a consumer advocacy websites may contain up to six steps referring to the same subject, as complainants are not willing to pursue their complaints any further if they have not reached a winning situation at that point.

[10] For space reasons we refer the interested reader to [22] for an in-depth discussion.

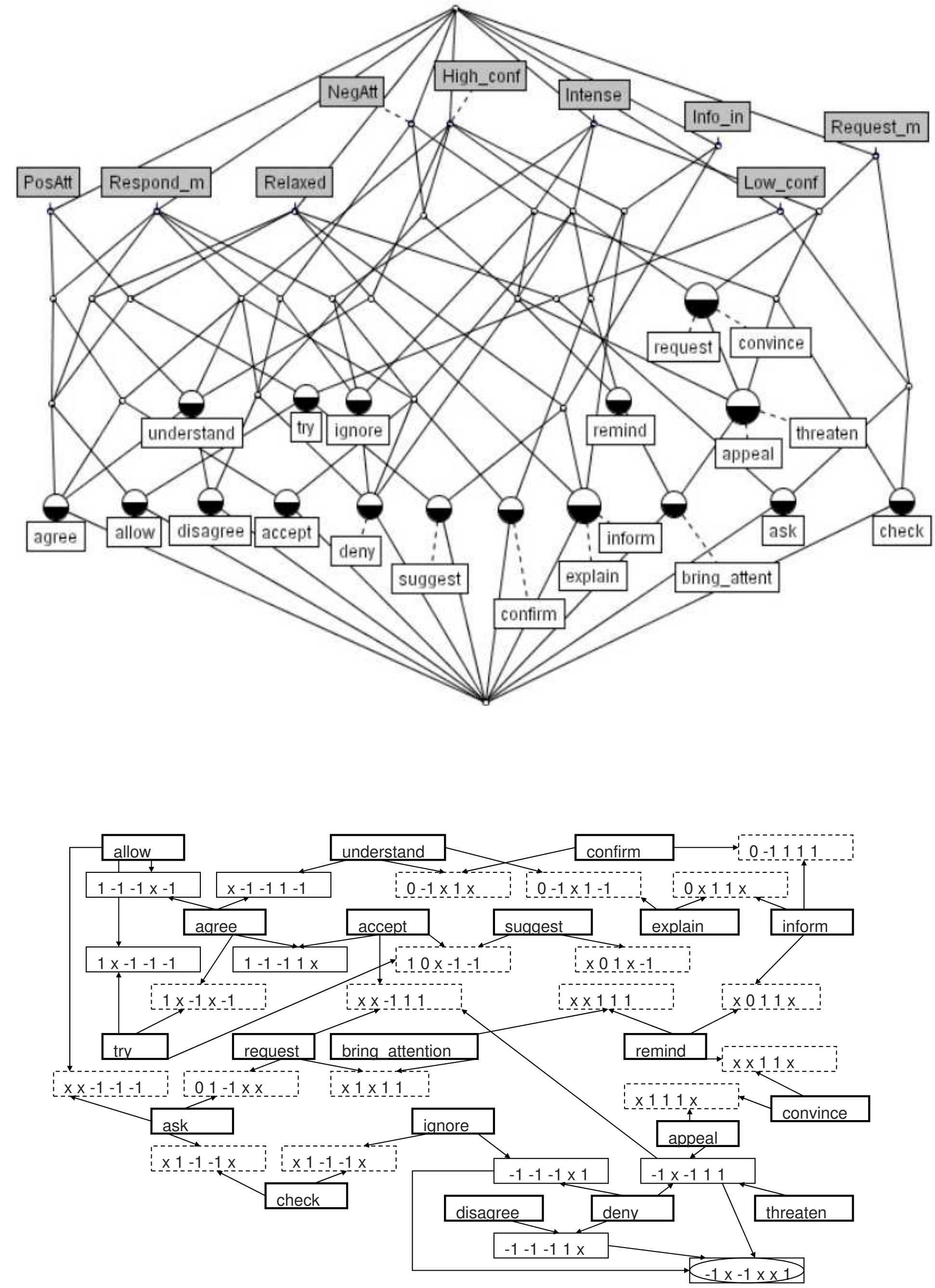

Figure 5: The concept lattice for communicative actions (top); Clustering communicative actions based on attributes (bottom)

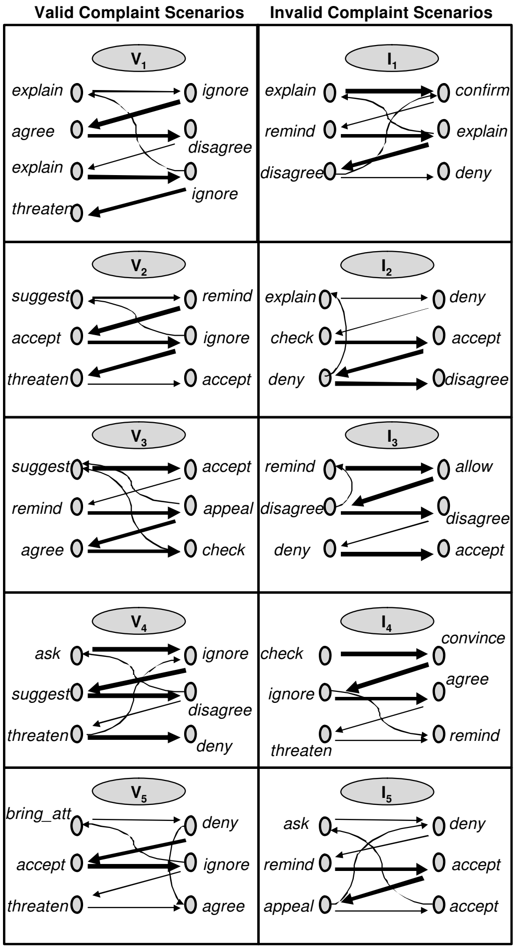

Figure 6: Sample training dataset of valid and invalid complaint scenarios

common subgraphs of C_X and C_Y, such that each graph $G_i \in C_X * C_Y = (V_i, A_i)$ is characterized as follows: i) v_i is a vertex in G_i iff v_i is a vertex in both C_X and C_Y which corresponds to communicative actions of the same party (customer or company); ii) (v_i, v_j) is a thick (resp. thin) arc in G_i iff (v_i, v_j) is a thick (resp. thin) arc in C_X and C_Y; iii) (v_i, v_j) is a thin arc in G_i iff (v_i, v_j) is a thick (resp. thin) arc in C_X and (v_i, v_j) is a thin (resp. thick) arc in C_Y; iv) (v_i, v_j) is an attack arc in G_i iff (v_i, v_j) is an attack arc in C_X and in C_Y; and v) G_i contains at least one thick arc (v_i, v_j). Note that when (v_i, v_j) is of the same type (thin or thick) in both C_X and C_Y, then that type is adopted for (v_i, v_j) in G_i (condition ii). Condition iii) specifies that a thin arc (v_i, v_j) is adopted as an arc in G_i whenever there are arcs (v_i, v_j) in C_X and C_Y of different types (thin arcs are seen thus as a weaker generalization of both thick and thin arcs). Attack arcs, on the other hand, are considered separately (condition iv). Finally, common subgraphs are required to have at least one thick arc (condition v).

By applying this definition of similarity we are now able to provide a criterion for relating a new complaint scenario to the class of valid/invalid scenarios, on the basis of its similarity with previous scenarios in the training dataset. We will assume that the training dataset $D = \{R_1, R_2, \ldots, R_n\}$ will contain both positive and negative examples, denoted as R^+ and R^-, respectively. In order to assign a new complaint scenario U to the class of "valid complaints" the following conditions should hold:[11]

1. U is similar to (i.e., has a nonempty common scenario subgraph of) some positive example R_i^+ in the training dataset D;
2. For any negative example R_j^-, if U is similar to R_j^- (i.e., $U * R_j^- \neq \emptyset$) then $U * R_j^- \subseteq U * R_i^+$.

The first condition requires that U is similar to some positive example in the training dataset D. The second condition establishes how to evaluate similarities when U can belong to both the positive and the negative class of examples. To be assigned to a class, the similarity between the unknown complaint scenario U and the closest scenario for the positive class should be higher than the similarity between U and each negative example corresponding to the class of invalid complaints. Note that condition (2) implies that there exists a positive example R_i^+ such that for no $R_j^- \in D$ it is the case that $U * R_i^+ \subseteq R_j^-$, i.e., there is no counterexample to this generalization of positive examples. Next we will show a worked example to illustrate how similarity is applied for classifying a new complaint scenario on the basis of a training dataset.

Example 2. *Consider the complaint scenario U in Fig. 7 (top). We want to determine whether U belongs to the class of valid complaints or to the class of*

[11]In our analysis, we restrict ourselves to the classification of a new scenario with respect to the positive class (i.e., valid complaints). The classification with respect to the negative class is made analogously.

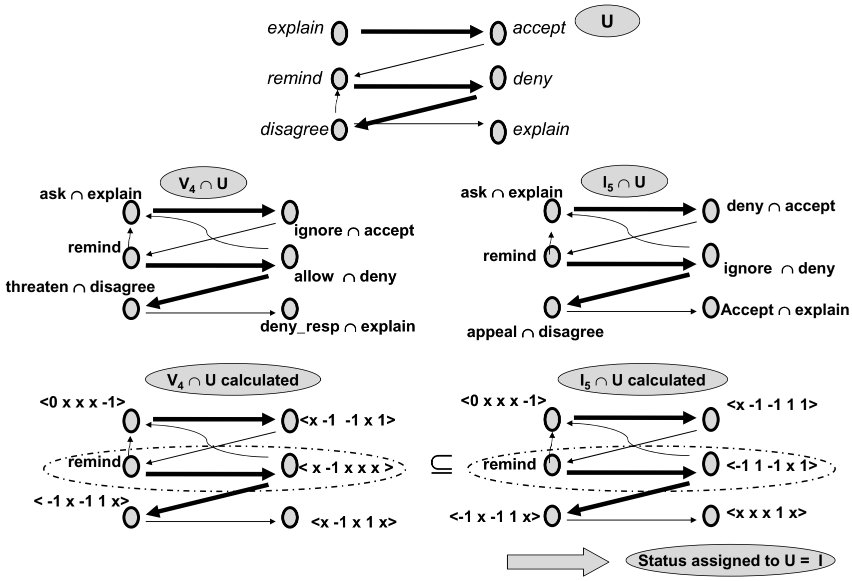

Figure 7: A scenario U with unassigned complaint status (top) and a sketch of the procedure for relating this scenario to a class. In this particular case, according to the available training data the assigned status for U is *Invalid*.

invalid complaints, according to the similarity criterion defined before and the training dataset given in Fig. 6.

Let us consider the set of training examples belonging to valid complaints. We can observe that V_4 is the graph with the highest similarity wrt U among all graphs from the set $\{V_1, \ldots V_5\}$ and find the singleton corresponding to the common subscenario $U * V_4$. The only thick arc in $U * V_4$ is derived from the thick arc between vertices with labels remind *and* deny *of U and the thick arc between vertices with labels* remind *and* allow *of V_4. The first vertex of this thick arc in $U * V_4$ is* remind $\cap$ remind $=$ remind, *the second is* allow $\cap$ deny $= < x\ -1\ x\ x\ x >$ *($U * V_4$ is calculated at the left bottom).*

Similarly, we analyze U wrt the set of training examples belonging to invalid complaints. We build the common subscenario $U * I_5$, as I_5 delivers the largest sub-graph (two thick arcs) in comparison with I_1, I_2, I_3, I_4. Moreover, we have that $U * V_4 \subseteq U * I_5$ (this inclusion is highlighted by the ovals around the steps), and for any other $V_i \neq V_4$, it holds that $U * V_i = \emptyset$, so that it trivially holds that $U * V_i \subseteq U * I_5$ and Condition 2 is satisfied. Therefore, U is classified as an invalid complaint, having the highest similarity wrt the invalid complaint I_5.

5. Evaluation

In order to test our approach, we used several textual complaints which were downloaded from the public website `PlanetFeedback.com` during three months starting from March 2004. For the purpose of this evaluation, each complaint was manually coded as a sequence of communicative actions, being assigned with a particular status by human experts. It must be remarked that not all complaints submitted by upset customers to consumer advocacy websites can be assessed with respect to validity. Clearly, the validity of those complaints just mentioning a failure in a product or a service without describing any interaction with the customer support cannot be assessed using the proposed technique.

This complaint preprocessing resulted in 560 complaints, divided in fourteen different banks or datasets, each of them involving 40 complaints. In each bank 20 complaints were used for training and 20 complaints for evaluation. During the evaluation, our goal was to analyze the accuracy of our proposal for classifying new customer complaints taking into account the underlying graph representation of complaint scenarios. This was done by extending the *ComplaintEngine* suite, incorporating the proposed similarity-based criterion as decision mechanism associated with complaint classification. The evaluation process involved three stages: in a first stage we evaluated the understandability and adequacy of the *ComplaintEngine* interactive form with respect to the users. In a second stage we applied supervised learning to classify 20 complaint scenarios out of the 40 complaint scenarios available in each of the 14 databanks provided. Finally, in a third stage, we contrasted the results obtained in our approach with respect to the previous methodology used in *ComplaintEngine* (classifying complaints only on the basis of the communicative actions used). In the next subsections we will discuss these different stages in detail.

5.1. Stage 1: Evaluating Understandability and Adequacy of the Complaint-Engine *Interactive Form*

To verify the adequacy of the proposed model we used the Interactive Form provided by *ComplaintEngine*, thus enforcing users to model their complaints in terms of communicative actions and attack links among them. The understandability and adequacy of the *ComplaintEngine* interactive form was evaluated by a team of individuals divided into three classes: complainants, company representatives and judges. Complainants had as a task to read a textual complaint and input it into the Interactive Form so that another team member (a company representative) could comprehend it (and briefly sketch the plot as a text). A third team member (judge) compared then the original complaint and the one written by the company representative as perceived from the Form. The result of this comparison was the judgment on whether the scenario structure has been distorted with respect to the validity of a given complaint.

Fig. 8(a) shows the results obtained in this first stage. It must be noted that only less than 15% of the complaints were hard to capture by means of communicative actions. We also observed that about 30% of the available complaints lost important details and could not be adequately restored (although they might still be properly related to a particular class). This situation happened with complex textual complaints which could not be properly modelled by means of the current *ComplaintEngine* interactive form. Nevertheless, the proposed formalization was adequate for most common textual complaints, covering a large number of cases (70%). Some alternatives for improving the representation adequateness of our approach are currently under consideration (see discussion in Section 7).

5.2. Stage 2: Evaluating the Classification Accuracy

In a second stage we evaluated the accuracy of the similarity-based approach for classifying complaint scenarios on the basis of the 560 complaints in the 14 databanks provided (40 complaints per databank). In each of the databanks two subsets were distinguished: a training set of 20 scenarios (classified as valid or invalid by human experts), and an evaluation set provided by the remaining 20 scenarios, which were unclassified.

Fig. 8(b) contains the results of our validity assessment, organized as follows: the first three columns contain dataset number and the numbers of valid/invalid complaints in each training set as manually assessed by human experts (two light-grayed columns on the left). The self-evaluation of training dataset column (third light-grayed column) shows the percentage of complaints that were correctly classified (with respect to the assessment of human experts) when the training dataset is used for *both* training and evaluation. [12] The middle area (Classification results) gives the number of complaints that were classified correctly and incorrectly in each databank using the similarity-based approach, as well as the number of false positives and false negatives obtained. We also distinguish the set C_{incons} (1st dark-grayed column) of those complaint scenarios in each databank which were classified inconsistently by our approach, belonging to both the class of valid and invalid complaints. Using the logic programming-based system Jasmine for machine learning (see [19] for details), we could identify those cases for each databank in the training set which were connected to deliver C_{incons}. After removing those cases, we performed a new evaluation on the set C_{incons} using our similarity-based approach. The results obtained for this new classification are shown in the 2nd, 3rd and 4th dark-grayed columns. Finally, the rightmost column (overall classification accuracy) gives the number of correctly assigned complaints as a percentage of the total number of complaints in the evaluation dataset.[13]

From our experiment we could conclude that the resultant recognition accuracy was 70%. Being quite low in accordance to pattern recognition standards in such domains as speech recognition [30], this accuracy is believed to be satisfactory for the decision-support settings where the number of complaints which have to be re-assessed manually is relatively low.

[12] This corresponds to the resubstitution error estimate [33] in which the training set and the evaluation set used are the same.

[13] This percentage is calculated as $(V + I + VR + IR)/20$, where V (resp. I) is the number of complaints classified as valid (resp. invalid) with respect to the training set, and VR (resp. IR) is the number of complaints classified as valid (resp. invalid) when a new evaluation on C_{incons} is performed.

Dataset (Bank #)	% of complaints understood and (somehow) specified in the Form	% of complaints (at least partially) reconstructed from the Form	% of complaints which were properly specified and reconstructed
# 1	85	75	65
# 2	80	75	60
# 3	95	85	75
# 4	90	85	75
# 5	90	80	75
# 6	80	75	70
# 7	85	75	65
# 8	90	85	75
# 9	85	75	65
# 10	85	80	70
# 11	90	80	65
# 12	85	75	65
# 13	80	70	60
# 14	90	85	75
Avg.	86.4	78.6	68.6

Bank #	As assigned by experts			Classification results					Evaluation on inconsistent classification			Overall classif. accuracy (%)
	Valid	Invalid	Self-eval. of training dataset (%)	Valid	Invalid	False positives (classified as valid but invalid)	False negatives (classified as invalid but valid)	Inconsistent classification (refuse to classify)	Valid	Invalid	Inconsistent or wrong cclassif.	
1	8	12	80	6	8	1	1	4	1	0	3	75
2	6	14	75	6	9	2	0	3	0	0	3	75
3	7	13	80	5	8	2	1	4	0	1	3	70
4	5	15	75	3	9	2	2	4	0	1	3	65
5	8	12	80	5	7	3	2	3	1	0	2	65
6	8	12	65	4	8	2	2	4	1	0	3	65
7	11	9	75	6	6	1	3	4	0	0	4	60
8	8	12	80	6	8	1	1	4	0	1	3	75
9	7	13	75	4	8	1	2	5	1	1	3	70
10	9	11	80	6	8	3	1	2	0	0	2	70
11	10	10	85	6	7	2	2	3	1	1	1	75
12	5	15	75	2	11	1	2	4	1	0	3	70
13	10	10	75	6	4	2	1	7	2	1	4	65
14	8	12	80	7	10	0	1	2	0	0	2	85
Avg.	7.9	**12.1**	**77.1**	**5.1**	**7.9**	**1.6**	**1.5**	**3.8**	**0.6**	**0.4**	**2.8**	**70.4**
%	39	**60.7**		**25.7**	**39.6**	**8.2**	**7.5**	**18.9**	**2.9**	**2.1**	**13.9**	

Figure 8: a) Representation adequateness for 14 databanks (left); b) Results obtained (right)

Graph $(V, A_{thick} \cup A_{thin})$ (communicative actions only):	64 %
Graph (V, A_{attack}) (attack links only):	43 %
Graph (V, A) (full graph) :	70.4 %

Figure 9: Recognition accuracy in classifying complaints with different underlying graph representations for complaint scenarios

5.3. Stage 3: Contrasting the Incidence of Communicative actions and Attack links

As a final stage in our evaluation, we performed a comparative analysis of relating complaint scenarios to the class of valid/invalid complaints. Given a complaint scenario $G = (V, A)$, with $A = A_{thick} \cup A_{thin} \cup A_{attack}$, as presented in Def. 2, we distinguished the three particular cases in our analysis:

(a) The graph $(V, A_{thick} \cup A_{thin})$ associated with the original representation used in the *ComplaintEngine* suite, involving only sequences of communicative actions (no attack relationships considered).

(b) The graph (V, A_{attack}), corresponding to a complaint scenario where temporal precedence between actions is not taken into account, and

(c) The full graph (V, A), including temporal precedence and attack relationships between communicative actions.

Figure 9 shows the recognition accuracy obtained for our dataset. We can see that the inclusion of attack links as part of the graph associated with the complaint scenario improves the classification accuracy for our dataset by about

22%, whereas the attack-only analysis delivers less than 50%. In such a setting, we can conclude that taking into account argumentation is significant for an accurate assessment of complaint validity. It must be remarked that we have intentionally limited our classification of complaints to valid and invalid, as these are two major classes with respect to how companies need to respond to these complaints. Additionally, we classified in our analysis only "mature" complaints (*i.e.,*those which include dissatisfaction with the product and also a follow-up interaction with the customer support), ignoring those which were not fully filled in by the user. As discussed before in the context of the evaluation experiments, these mature complaints include sufficient data to judge their validity by applying our proposed approach.

6. Related work

As discussed in the introduction, there are several software tools [26, 31, 1, 11] oriented towards providing automated customer complaint management systems. However, to the best of our knowledge, there is no similar approach to classifying customer complaints using supervised learning as proposed in this paper. Indeed, there have been some applications of machine learning techniques to identify complaint situations, as done in [42]. In this paper, the authors apply rough set theory to discover relevant attributes which might lead to complaints in packaging foundry for integrated circuits. In contrast with our approach, they induce decision rules aiming to identify the offending attributes, rather than analyzing the validity of customer complaints, as presented in this paper.

The use of dialogical argumentation as underlying element for modelling decision making situations as those associated with complaint processing is not new. In [40], a conceptual framework for DSS based on critique and argumentation is presented, where the use of debate and argumentation are proposed as means for more informative decision support. The architecture of the proposed DSS contains intelligent critiquing agents which provide the user with the qualitative feedback on candidate decisions. In contrast with our approach, this framework is rather generic, and does not rely on machine learning techniques for answering user's requests. Argumentation has also been applied in the context of decision making in business contexts, as done by [37, 38]. In contrast with our formalization, this approach allows for assessments over a continuum (rather than a binary "valid - invalid" evaluation), resulting in a better perspective for decision making in business applications. Recent research has also been oriented towards developing dialogical systems (e.g., [32, 12]) which have simple speech act constitutive elements, and have been shown to be complete to formalize many negotiation tasks. In spite of their expressivity, such systems cannot be directly adapted to complaint situations, as we focus on a negotiation between human agents (company and customer), presented in restricted natural language (rather than in a specialized formal language as in dialogical systems), augmenting our inference capabilities by means of supervised learning.

There exists a number of settings in which graph-based datamining and clustering is performed (e.g., [7, 28]) relying on information-theoretic or error-based measures. The usual technique applied in such settings is to find a subgraph S that appears often in the positive graphs G^+, but not in the negative graphs G^-, minimizing the equation $\frac{|\{g \in G^+ | S \not\subseteq g\}| + |\{g \in G^- | S \subseteq g\}|}{|G^+| + |G^-|}$, where $S \subseteq g$ means that S is isomorphic to some subgraph in g. When delivering results, a learning procedure based on the above measure would back up its decision with respect to a threshold value. In contrast with this approach, our supervised learning algorithm backs up its decisions with an enumeration of similar objects and their features, providing thus a more qualitative analysis for both customers and company representatives.

Finally, it is interesting to note that the usability of the underlying representation machinery for scenarios of inter-human interactions goes beyond the domain of customer complaints. In part of our recent research [20, 16, 17] five different domains were considered to assess the adequateness of speech act theory, obtaining satisfactory results. Such domains included international conflicts, security clearance scenarios, detection of emotional profiles, analysis of bloggers' sentiments and identification of suspicious behavior of cell phone users. This provides an empirical support for the adequacy of our graph-based representation language involving communicative actions characterized by numerical-valued attributes.

7. Conclusions. Future work

Processing customer complaints is a major challenge in the context of knowledge management technologies nowadays. In this paper we have proposed a novel approach to improve automated processing of customer complaints. We have shown how communicative actions and attack links can be successfully modelled in terms of the graph-based representation provided by the notion of complaint scenario. We have also shown that our proposal for classifying complaint scenarios using supervised learning can be successfully applied, outperforming the results obtained using the *ComplaintEngine* platform when applying a keyword-based approach in which no attack links were taken into account for complaint classification. In this respect, the evaluation experiments using our dataset of formalized real-world complaints showed a satisfactory performance. It must be remarked that a particular strength of our approach is that the set S_{freq} and the associated concept lattice were computed once, accounting for the whole domain of complaint scenarios. Clearly, for other application domains associated with conflicting situations are involved (e.g. international conflicts), the corresponding set S_{freq} of communicative actions and consequently the associated concept lattice could be different. Part of our current research is oriented towards the study of communicative actions in other application domains [20, 16, 17].

Recent research in the context of decision support systems has been oriented towards the development of argument-based extensions of logic programming. Defeasible Logic Programming (DeLP for short) [24] is one of such extensions, and has been successfully applied in the context of decision making for knowledge management [5]. Part of our current work is oriented towards considering the integration of DeLP as a way of developing new, enhanced matching mechanisms for classifying complaint scenarios [18]. In this respect, DeLP would provide a powerful tool for formalizing knowledge and performing automated defeasible reasoning from the underlying graph structure associated with complaint scenarios. However, further experimentation is still needed in this direction in order to determine whether the success rate for classifying customer complaints we have obtained with the *ComplaintEngine* (Section 5) can be improved by incorporating these new features. The integration of DeLP has also motivated us to study how to improve the current *ComplaintEngine* Interactive Form in order to include more possibilities to model argument-based attacks. We think that DeLP could help to improve the representation adequateness of the *ComplaintEngine* when modelling those complex complaints which could not be captured in the current version of the system, as discussed in Section 5.1.

As a final conclusion we can say that the proof-of-concept evaluation of our approach shows that it is a promising technique to classify customer complaints integrated in the context of a major software infrastructure for decision support in complaint processing. Research in this direction is currently being pursued.

Acknowledgments

We want to thank the reviewers for their fruitful comments and suggestions which helped to improve the contents and structure of this article. Carlos Chesñevar was supported by projects TIN2006-15662-C02-01 (MEC, Spain), Project 24/ZN10 (SGCyT, UNS), by CONICET (Argentina) and by Ramón y Cajal Program (MCyT, Spain). María Paula González was supported by CONICET (Argentina) and by project TIN2008-06596-C02-01 (MEC, Spain). We thank Sergei Kuznetsov and Mikhail Samokhin for fruitful discussions.

References

[1] Amadeus. Amadeus compliance process control suite. `http://www.amadeussolutions.com/english/practices/bp_customer_complaint.htm`.

[2] J. L. Austin. *How To Do Things with Words.* Oxford University Press, Oxford, UK, 1962.

[3] K. Bach. *Routledge Encyclopedia of Philosophy*, chapter Speech Acts. Routledge, 1998.

[4] K. Bach and R. Harnish. *Linguistic Communication and Speech Acts.* Cambridge (MA): MIT Press, 1979.

[5] R. Brena, J. Aguirre, C. Chesñevar, E. Ramirez, and L. Garrido. Knowledge and information distribution leveraged by intelligent agents. *Knowledge and Information Systems*, 12(2):203–227, 2007.

[6] Y. Cho, I. Im, R. Hiltz, and J. Fjermestad. An analysis of online customer complaints: Implications for web complaint management. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS'02)-Volume 7*, pages 2308–2317, 2002.

[7] J. Coble, D. Cook, R. Rathi, and L. Holder. Iterative structure discovery in graph-based data. *Intl. Journal of Artificial Intelligence Techniques*, 1-2(14):101–124, 2005.

[8] CWA. Cwa flow / complaint management. `http://www.cwa.eu/en/solution_cwa_flow.html`.

[9] M. Davidow. Organizational responses to customer complaints: What works and what doesn't. *Journal of Service Research*, 5(3):225–250, 2003.

[10] W. Duana, B. Gub, and A. Whinston. Do online reviews matter? – an empirical investigation of panel data. *Decision Support Systems*, 45(4):1007–1016, 2008.

[11] I. EtQ. Complaint handling software. `http://www.etq.com/complaint/index.shtml`.

[12] R. A. Flores, P. Pasquier, and B. Chaib-draa. Conversational semantics sustained by commitments. *Autonomous Agents and Multi-Agent Systems*, 14(2):165–186, 2007.

[13] B. Galitsky. A tool for extension and restructuring natural language question answering domains. *Lecture Notes in Computer Science (Procs. of the 15th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems IEA/AIE 2002 Cairns, Australia, June 1720, 2002).*, 2358:223–255, 2002.

[14] B. Galitsky. A library of behaviors: Implementing commonsense reasoning about mental world. volume 3215, pages 307–313. Springer, 2004.

[15] B. Galitsky. Reasoning about attitudes of complaining customers. *Knowledge-Based Systems*, 19(7):592–615, 2006.

[16] B. Galitsky and B. Kovalerchuk. Mining the blogosphere for contributors' sentiments. In *Proc. of the AAAI Spring Symposia on Computational Approaches to Analyzing Weblogs (published as technical report). Stanford, CA, USA*, 2006.

[17] B. Galitsky, B. Kovalerchuk, and S. Kuznetsov. Learning common outcomes of communicative actions represented by labeled graphs. In *Proc. of the 15th Intl. Conference on Conceptual Structures (ICCS). Sheffield, UK*, pages 387–400, July 2007.

[18] B. Galitsky, S. Kuznetsov, and B. Kovalerchuk. Argumentation vs meta-argumentation for the assessment of multi-agent conflict. In *Proc. of the Workshop on Metareasoning (colocated with the 23rd AAAI Conference 2008), published as Technical Report*, 2008.

[19] B. Galitsky, S. O. Kuznetsov, and D. V. Vinogradov. Applying hybrid reasoning to mine for associative features in biological data. *Journal of Biomedical Informatics*, 40(3):203–220, 2007.

[20] B. Galitsky and A. Miller. Determining possible criminal behavior of mobile phone users by means of analyzing the location tracking data. In *Proc. of the AAAI Spring Symposia on Homeland Security (published as technical report). Stanford, CA, USA*, 2005.

[21] B. Galitsky and I. Tumarkina. Justification of customer complaints using emotional states and mental actions. In V. Barr and Z. Markov, editors, *Proceedings of the Seventeenth International Florida Artificial Intelligence Research Symposium (FLAIRS) Conference, Miami Beach, Florida, USA*, pages 239–244. AAAI Press, 2004.

[22] B. Ganter and S. Kuznetsov. Hypotheses and version spaces. In *Proc. 10th Int. Conf. on Conceptual Structures (ICCS 2003). In A. de Moor, W. Lex, and B.Ganter, Eds., LNAI vol. 2746, pp. 83-95 (2003).* Springer, 2003.

[23] B. Ganter and R. Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer Verlag, 1999.

[24] A. García and G. Simari. Defeasible Logic Programming: An Argumentative Approach. *Theory and Practice of Logic Programming*, 4(1):95–138, 2004.

[25] N. Glance, M. Hurst, K. Nigam, M. Siegler, R. Stockton, and T. Tomokiyo. Deriving marketing intelligence from online discussion. In *KDD '05: Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 419–428, New York, NY, USA, 2005. ACM.

[26] GTF. Gtf inspire complaint processing. `http://www.gft.de/software/en/start/solutions/complaint.html`.

[27] C. L. Hamblin. *Fallacies*. Methuen, London, 1970.

[28] L. Holder, D. Cook, J. Coble, and M. Mukherjee. Graph-based relational learning with application to security. *Fundamenta Informaticae (Special Issue on Mining Graphs, Trees and Sequences)*, 1-2(6):83–101, 2005.

[29] S.-Y. Hung, Y.-C. Ku, T.-P. Liang, and C.-J. Lee. Regret avoidance as a measure of dss success: An exploratory study. *Decision Support Systems*, 42(4):2093–2106, 2007.

[30] F. Jelinek. *Statistical methods for speech recognition.* MIT Press Cambridge, MA, USA, 1998.

[31] MasterControl. Mastercontrol: Customer complaint handling software. `http://www.mastercontrol.com/solutions/complaints_fb.html`.

[32] J. McGinnis, D. Robertson, and C. Walton. Protocol synthesis with dialogue structure theory. In *Proc. of EUMAS 2005. Brussels, Belgium*, pages 179–189, Dec. 2005.

[33] T. Mitchell. *Machine Lerning.* McGraw-Hill, 1997.

[34] K. Nigam and M. Hurst. Computing attitude and affect in text: Theory and applications. chapter Towards a Robust Metric of Polarity, pages 265–277. Springer, Dordrecht, The Netherlands, 2006.

[35] Oracle. Oracle customer relationship management. `http://www.oracle.com/applications/peoplesoft/crm/ent/index.html`.

[36] Oracle. Oracle text. `http://www.oracle.com/technology/products/text/index.html`.

[37] T. S. Raghu, R. Ramesh, A. B. Whinston, and A.-M. Chang. Collaborative decision making: A connectionist paradigm for dialectical support. *Information Systems Research*, 12(4):363–383, 2001.

[38] R. Ramesh and A. B. Whinston. Claims, arguments, and decisions: Formalisms for representation, gaming and coordination. *Information Systems Research*, 5(3):294–324, 1994.

[39] J. Searle. *Speech Acts: An Essay in the Philosophy of Language.* Cambridge University Press, Cambridge, UK, 1969.

[40] R. Vahidov and R. Elrod. Incorporating critique and argumentation in dss. *Decision Support Systems*, 26(3):249–258, 1999.

[41] D. Walton. *A Pragmatic Theory of Fallacy (Studies in Rhetoric and Communication Series).* The University of Alabama Press, Tuscaloosa, 1995.

[42] H.-H. Yang, T.-C. Liu, and Y.-T. Lin. Applying rough sets to prevent customer complaints for ic packaging foundry. *Expert Systems with Applications*, 32(1):151–156, 2007.

[43] S. Yevtushenko. Conexp software. `http://www.sf.net/projects/conexp` (Last accessed Jan 2005)

[44] A. Yuksel, U. Kilinc, and F. Yuksel. Cross-national analysis of hotel customers' attitudes toward complaining and their complaining behaviours. *Tourism Management*, 27(1):11–24, 2006.