INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'07
28 - 31 AUGUST 2007, CITÉ DES SCIENCES ET DE L'INDUSTRIE, PARIS, FRANCE

HOW AM I DOING 2: COMPUTING THE LANGUAGE OF APPRAISAL IN DESIGN

Jianxiong Wang and Andy Dong
Key Centre of Design Computing and Cognition, University of Sydney

ABSTRACT
There has been a wide body of design research about the categorisation of design text by the content or role of the text. However, recent research on information use in design has suggested the need to consider the semantic orientation toward the subject matter because the orientation might signal design rationale and other critical meta-functions. This paper presents a computational implementation of a natural language classifier for the understanding of appraisal in design text, that is, the expression of an opinion or attitude toward the designed work, the design process, and the design team. Using the supervised machine learning technique of support vector machines and data from informal and formal design documents, including transcripts of conversations between designers, we find that machine learning techniques can be trained to categorise the semantic meaning of design documents and the orientation of the opinion expressed in the document.

Keywords: Knowledge management, language in design

INTRODUCTION
There has been a wide body of design research concerning the categorisation of design text by understanding the content or function of the text. Much of this research was conducted for purposes such as design rationale capture, decision support, and case-based reasoning. However, recent research on information use in design has suggested the need to consider the semantic orientation (i.e., the opinion, sentiment, or attitude expressed) toward the subject matter within the text in addition to knowing the semantic meaning of the text. To illustrate why semantic meaning is not sufficient to fully understand design text, let's consider the following exchange between the participants of the backpack design team from the 1994 Delft Protocols Workshop [1].

Kerry (t 558) ***seems like lower is better regardless*** *as you say like we design in the low position and* ***not necessarily try and get***
John (t 627) if it was a smaller article it would work but ***not if it's something this size*** *um … and over the front does do people have any problems with mounting it up front?*
Ivan (t 628) yeah

In this section of the conversation, the content (semantic meaning) is about ergonomic issues. However, their attitude toward the placement of the backpack is contained in the appraisal "seems like lower is better regardless". This appraisal also signals their design rationale, which is then supported by an explanation of the rationale in the clauses "not necessarily try and get", "if it was a smaller article it would work", "but not if it's something this size", and "you have more mass up there to turn" later on, all of which are forms of appraisal. Other meta-functions of appraisal include negotiation or shifting the evaluation of a designed work from technical analysis to subjective consideration. While these meta-functions might occur in design briefs (function of the text) about a specific attribute of the designed work (content), these meta-functions are registered in appraisals. The point here is to distinguish between semantic meaning (what a design document is about) and semantic orientation (the attitude of the author to the subject matter of the design document).
In a prior paper [2], we modelled the language of appraisal in design, adapted the technique of functional grammatical analysis to analyse the various ways that appraisal is registered in language, and applied the technique to various design texts to characterise the meta-functions of the language of appraisal in design. In this paper, we present an alternative method for understanding the language of

appraisal in design using a computational natural language classifier based on statistical machine learning. Given the model of the language of appraisal in design presented in our prior work, the classifier categorises the design text as being about Product, Process or People, and then classifies the associated semantic orientation.
The next section briefly reviews the theory of appraisal and the theory of the language of appraisal in design. Given the model, we then present a computational model for the language of appraisal in design. This model is used to encode design text to enable a natural language classifier to categorise and classify the text according to semantic meaning and semantic orientation. We show and discuss the performance results of the system and discuss the types of studies and design text applications that such a text classifier enables.

APPRAISAL

In linguistics, studies of pragmatics focus on how people apprehend and generate a communicative act or speech act in a concrete speech situation such as a conversation. Pragmatics considers ideas as instruments that function as guides of action, their validity being determined by the success of the action. Analysis of pragmatics in communication separates meanings or intents in each talking session or verbal communication into two components [3]. One is the informative purport or what the sentence is getting at, and the other the emotional tendency or speaker intent [4]. Appraisal (semantic orientation) focuses only on how human emotion is expressed in language.
Martin [5] defines appraisal as the semantic resources used to negotiate emotions, judgments, and valuations, alongside resources for amplifying and engaging with these evaluations. While intimately coexisting with the content of a clause, the function of appraisal is to negotiate attitudinal stances toward the content. The goal of research in appraisal in linguistics is to discover systematic approaches to derive reliable linguistic resources that deal with both the representation and the representability of feeling through language and the function of grammar in enacting appraisals. The theory of appraisal identifies five resources (high-level options) for conveying an appraisal: attitude, engagement, graduation, orientation and polarity. Attitude gives the type of appraisal expressed which is either affect (relating to emotional states), appreciation (of objects), or judgment (of agents). Engagement is the commitment to the appraisal and is often considered an appraisal of the appraisal. It deals with subtle grading of the speaker's commitment to what is expressed. Graduation deals with the strength of the evaluation. Orientation relates to whether the appraisal is positive or negative. Polarity is labelled as marked or unmarked depending upon whether the appraisal is scoped. To illustrate the registration of these components of appraisal in design text, let us take the following example from the Delft protocol as above. In this statement, the group is exploring the rack to bike to connection.

*John (t 989) does it **do we really wanna use these lugs for speed of disassembly or does it make more sense to like just have something** that like a plastic ferrule **or something** that goes around this that you*

Let us focus specifically on the clause "do we really want use these lugs for speed of disassembly". The function of the verb "wanna" is to express the attitude (appreciation). John strengthens his appreciation (negative orientation) toward the lugs through the technique of graduation by saying "**really** wanna". The negative orientation is signalled through the technique of comparison with the word "or". Finally, John uses the technique of quantification in amount to increase the force of the appraisal by offering two other possibilities to graduate his dislike for the lugs. Rhetorically, the repetition of "or" twice strengthens both the force and engagement with the appraisal.
However, there exist some theoretical problems with the general linguistic theory of appraisal. Since appraisals are evaluative, it becomes important to know what object the appraisal is directed toward. This is consistent with the object-relations theorists' view on the link between appraisals and emotions. Their view is that emotions arise from value judgments (appraisals) ascribed to objects and persons outside of a person's own control and which are of importance for the person's flourishing [6]. That is, an emotion is always intentionally directed at an object. The object-relations theorists construe the concept of an object broadly to include agents, things, and events. To them, an appraisal is always intentionally directed at an object. Second, the notion of judgment in the general theory of appraisal is limited to agents only. In design, judgments of objects and activities are routine; they are based on norms such as schools of design, accepted practices, and design requirements. Objects are judged in design and not just subjectively appreciated as the SFL theory prescribes. Finally, affect is too broad

of a grouping and does not adequately distinguish between affect, cognitive, and behavioural dispositions in line with relevant research in cognitive science on the human affective system [7]. It also does not deal with assessment of capabilities to undertake design activities.
In light of these problems, we developed a model of the language of appraisal in design [2]. At the top level of this model are a set of linguistic resources to accomplish reflection-in-action and attitudinal positioning toward Product, Process and People. From these, designers or design text can express its Attitude toward Product, Process and People through: 1) affect – how the designer appraises affective, cognitive and behavioural conditions that represent how the designer is thinking as well as how the designer is behaving in a cognitive-behavioural sense [7]; 2) judgment – how the designer appraises in relation to the accepted norms such as standards, industry best practices or normative design methods, and objective criteria established by the design brief; 3) appreciation – how the designer appraises in relation to personal experience (e.g., expertise and intuition) and subjective interpretations; and 4) capability – how the designer appraises capability to or functioning of a person doing a design-related activity. The model leaves graduation, orientation and polarity unchanged from the general linguistic theory.
This paper focuses on a computational implementation of two resources in the language of appraisal of design, the category of the appraisal and the orientation of the appraisal, and their attributes, product, process or people, and positive or negative, respectively. This abridged model of the language of design is depicted in Figure 1 and sets the groundwork for our computational implementation of the language of appraisal in design.

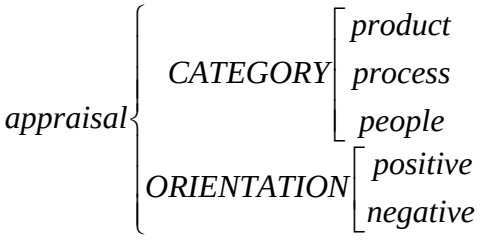

Figure 1. Linguistic resources (abridged) for appraisal and their attributes

SENTIMENT ANALYSIS

The semantic orientation of an appraisal continues to form the subject matter of vigorous research in the computational linguistics community. In computational linguistics, the analysis of appraisal in text is called sentiment analysis. Both statistical natural language processing (NLP) and machine learning have been applied to the problems of identifying whether text contains subjective content and then the orientation of the subjective content. Statistical NLP methods were the first ones attempted toward these problems. Running a probabilistic classifier on POS (part-of-speech), lexical and formatting features, Wiebe [8] developed a model that predicted subjective sentences with 72.1% accuracy. The research team also showed that a sentence is 55.8% likely to have subjective content if there is an adjective within. To date, the best result for classifying the subjective material has come from Hatzivassiloglou and McKeown [9]. They combined a log-linear statistical model that examined the conjunctions between adjectives, such as “and”, “but”, “or”, with a clustering algorithm that grouped the adjectives into two sets which were then labelled positive and negative. Their model predicted whether adjectives carried positive or negative sentiment with 82% accuracy. However, because the model was unsupervised it required an immense 21 million word corpus to function. The limitation to adjectives is problematic since nouns (such as “masterpiece”) and verbs (such as “prefer”) are often employed to evoke an appraisal.
Turney [10] matched bigrams of part-of-speech units and applied a Pointwise Mutual Information – Information Retrieval (PMI-IR) method to rate four domains of reviews: automobiles, banks, movies and travel destinations. The PMI-IR equation was based on the dependence of two words by dividing the probability of co-occurrence by the probability that they co-occur independently, which was compared against the words “excellent” and “poor”. Turney achieved an average of 74% accuracy across all domains against authors’ designations, but the performance on movie reviews was especially poor at only 65.8%. Turney mentioned that the low performance on movie reviews in particular was probably due to the non-compositional nature of such reviews [11].

More recently, machine learning has been identified as a robust mechanism for sentiment analysis because identifying the high-level options and attributes of appraisal could be handled as classification problems. A classifier could be trained through supervised learning to identify the resources for appraisal rather than just the subjective modifier terms as with statistical NLP methods. Once the resources have been identified, the attributes of those resources could be identified in a second pass. For example, the Orientation of an appraisal is one of two possibilities, positive or negative, and could thus be treated as a two-class classifier problem. (A neutral orientation is not considered an appraisal.) Many existing classification methods can thus be adopted for identifying the orientation of an appraisal and numerous attempts have been made including probabilistic classifiers (e. g. Naïve Bayes), linear classifiers, (e.g. support vector machines), and other AI-based methods such as neural networks (ANN).

Pang [12] showed that the task of classifying orientation in sentiment analysis was, at least from the viewpoint of machine learning, the same problem as topic classification, where the topic categories were "positive" and "negative." They compared Naïve Bayes, Maximum Entropy Classification, and Support Vector Machine classification techniques, machine learning methods known to be successful at topic classification tasks, to the Orientation of movie reviews. They reported that the Naïve Bayes method returned a 77.3% accuracy using bigrams as features against a human-generated list of likely positive and negative adjectives. Additionally, they calculated the presence and frequency of unigrams, the presence of bigrams, unigrams coupled with part of speech, adjectives, most frequent unigrams, and unigrams coupled with their position within the document. The best results came from unigram presence with Orientation classified by a support vector machine, achieving 82.9% accuracy against human classification of adjectives. Maximum Entropy Classification performed best using unigram and bigram presence at 80.8% accuracy, and Naïve Bayes performed best at 81.0% using just unigram presence. The results of all the tests performed hovered right around 80% on average.

The problem of distinguishing orientation can thus be considered as a two-class classification problem. Given the superior performance of support vector machines in both text categorisation problems [13] and sentiment analysis, we have opted to use support vector machines for classifying both the category of text and the orientation of the appraisal. The challenge in building a sentiment classifier for the language of design, however, is to produce a representation of the text that can be used to both classify the category of the text (Product, Process and People) and the semantic orientation rather than using a separate representation for each problem as is currently practiced. Second, the classifier must be able to perform over formal (e.g., memos, reports, briefs) and informal (e.g., conversations, e-mail) design text to be useful. The computational model developed in the next section responds to these challenges.

COMPUTATIONAL LANGUAGE MODEL OF APPRAISAL IN DESIGN

In order to make use of the support vector machines, we need to represent the design text in a vector format, preferably compact, that represents both the topic of the text and the sentiment of the text. The unigram and bigram structures of the Pang study cited above are insufficient for our purposes. Since a text might contain multiple sentiments containing all three categories, we will need to deal with smaller segments of text. In linguistics, the clause often serves as the unit of analysis. A clause is a sequence of words that express an action or a state. A clause must contain a verb which expresses the action and the participants in that action. A sentence can contain more than one clause. As one would expect, a sentence could have two separate categories and sentiments as with *Intel rapidly designs chips, but most of the new chips are minor incremental improvements* in which the first clause *Intel rapidly designs chips* comments on the pace of Intel's (Category: People) ability to design microprocessors and the second clause *most of the new chips are minor incremental improvements* comments on the microprocessors themselves (Category: Product). For each clause, we would like to encode which categories the words might belong to and whether the words express a positive or negative sentiment. It should be noted here that the expression of sentiments is not restricted to adjectives alone. Sentiments can be expressed in nouns, as with masterpiece as in *This is a masterpiece*, or in verbs, as with *dislike* as in *I dislike this concept*. Thus, for each clause, we need to encode the category – Product, Process, or People – and the sentiment of the clause's constituents, the verb and the participants in the clause including the noun and the adverb/adjective modifiers.

To encode this information requires $3 \times 3 = 9$ combinations, 3 dimensions for each category and 3 dimensions for each constituent in a clause. In total, we have a 9–dimensional vector of the following form:

$$\begin{bmatrix} Pd_N & Pd_V & Pd_A & \mathrm{Pr}_N & \mathrm{Pr}_V & \mathrm{Pr}_A & Pp_N & Pp_V & Pp_A \end{bmatrix}$$

Figure 2. 9-dimensional vector representation of text

where Pd = Product, Pr = Process, Pp = People, N = Noun, V = verb, and A = adjective/adverb.
The value of each of the vector dimensions is determined in the following way. First, we count the frequency of occurrence of a word in the target clause. Then, each word is looked up in the WordNet lexicographer database to ascertain the logical grouping that might indicate the appropriate CATEGORY for the word. The WordNet lexicographer database and their syntactic category and logical groupings were used to categorise words (nouns) as being about Product, Process or People. Verbs are categorised according to the category(ies) of the first Participant in the clause. The frequency of occurrence for all possible categories is placed in the appropriate position in the vector.
The following rules were applied to identify which of the WordNet logical groupings would contain nouns in the categories Product, Process and People. Following each of the rules are the names of the WordNet lexicographer files and the specification for the logical grouping of words in the file.

Any grouping which has to do with a material or performance attribute of the designed work is a **product** grouping.

noun.animal (nouns denoting animals) e.g. virus
noun.artifact (nouns denoting man-made objects) e.g. additive
noun.attribute (nouns denoting attributes of people and objects) e.g. shiftlessness
noun.cognition (nouns denoting cognitive processes and contents) e.g. inaptitude
noun.communication (nouns denoting communicative processes and contents) e.g. transmission
noun.food (nouns denoting foods and drinks) e.g. candy corn
noun.group (nouns denoting groupings of people or objects) e.g. amalgam
noun.location (nouns denoting spatial position) e.g. apogee
noun.object (nouns denoting natural objects (not man-made)) e.g. river
noun.plant (nouns denoting plants) e.g. pineapple
noun.possession (nouns denoting possession and transfer of possession) e.g. acquisition
noun.process (nouns denoting natural processes) e.g. advection
noun.quantity (nouns denoting quantities and units of measure) e.g. modulus
noun.relation (nouns denoting relations between people or things or ideas) e.g. gradient
noun.shape (nouns denoting two and three dimensional shapes) e.g. cylinder
noun.state (nouns denoting stable states of affairs) e.g. transparency
noun.substance (nouns denoting substances) e.g. vinyl

Any grouping that is about any actions and doings, both physical and cognitive, or the period of actions that could be taken by human or artificial agents toward the realization of the product is a **process** grouping.

noun.act (nouns denoting acts or actions) e.g. incitation
noun.cognition (nouns denoting cognitive processes and contents)
noun.event (nouns denoting natural events) e.g. accident
noun.feeling (nouns denoting feelings and emotions) e.g. relief
noun.motive (nouns denoting goals) e.g. incitement
noun.phenomenon (nouns denoting natural phenomena) e.g. photoconductivity
noun.process (nouns denoting natural processes)
noun.state (nouns denoting stable states of affairs)
noun.time (nouns denoting time and temporal relations) e.g. continuum

Any grouping that characterises humans is a **people** grouping.

noun.attribute (nouns denoting attributes of people and objects) e.g. shiftlessness
noun.body (nouns denoting body parts) e.g. person
noun.feeling (nouns denoting feelings and emotions)
noun.group (nouns denoting groupings of people or objects)
noun.person (nouns denoting people) e.g. designer
noun.relation (nouns denoting relations between people or things or ideas)
noun.state (nouns denoting stable states of affairs)

Two problems arise for which we need to apply correction factors. When the specification for a logical grouping indicates that the grouping will likely contain nouns which belong to more than one category, such as *noun.attribute*, *noun.relation* and *noun.state*, the logical grouping is placed in all relevant categories. In this case, the weight that is given to a logical group is inversely proportional to the number of possible categories. For example, because *noun.state* contains words across all three categories, its correction factor κ_1 is ⅓. The correction factor κ_1 is applied to a logical grouping in calculating a correction factor κ_2 for each word, which we describe next.

Often, a word will appear in multiple WordNet logical groupings and as such may have more than one possible category. For example, the word *right* appears in the logical groupings *noun.attribute* (People or Product), *noun.possession* (Product), *noun.location* (Product), *noun.act* (Process), *noun.group* (People), *noun.body* (People), and *noun.artifact* (Product) in this order of frequency of occurrence of the number of senses of the word *right* in the WordNet tagged texts. In this case, a word is counted as being in a category in proportion to the number of WordNet logical groupings in which it appears. This proportional weighting in calculating κ_2 as shown in Equation 1 captures the intuition that the actual category of a word is not as strongly known as a word which is grouped in a WordNet logical grouping under a single category. The frequency of occurrence of a word in the clause is thus multiplied by a correction factor κ_2. The intuition behind this correction factor is to take into account the uncertainty of a word's category.

$$\kappa_2(word) = \frac{\sum_{WordNetLogicalGroup \in category} senses_{WordNetLogicalGroup}(word) \times \kappa_1(WordNetLogicalGroup)}{\sum_{WordNetLogicalGroup} senses_{WordNetLogicalGroup}(word) \times \kappa_1^{-1}(WordNetLogicalGroup)} \quad (1)$$

Since the correction factor κ_2 for a word may have up to three values, it is normally expressed as a vector of the form $\kappa_2(\text{word}) = [\kappa_{2,Pd}, \kappa_{2,Pr}, \kappa_{2,Pp}]$.

Even though this initial categorisation and set of correction factors may not be precise, that is, words that occur in the category People might actually be a word about Product, the categorisation is likely to be accurate. That is, the category People will contain all possible words in the WordNet lexicon that deal with people. It should be possible to engineer the support vector machine with prior information about word categories and then let the reinforcement learning adjust the hyperplane to separate the categories more precisely given their occurrence in the training data set about design.

Second, the frequency count and correction factors are multiplied by a numerical value which calculates the semantic orientation of the term using Turney's Semantic Orientation – Pointwise Mutual Information (SO–PMI) metric which measures the strength of word co–occurrence for a given word for which we would like to calculate the degree of positiveness or negativeness with a known positive/negative word (pword or nword, respectively). The derivation of SO-PMI is based on the Pointwise Mutual Information – Information Retrieval (PMI–IR) metric which calculates the probability of occurrence of two words given statistical data in a very large corpus. The value of PMI–IR is calculated using Equation 2:

$$PMI - IR(word_1, word_2) = \log_2\left(\frac{p(word_1 \,\&\, word_2)}{p(word_1)p(word_2)}\right) \quad (2)$$

Web-based search engines provide a useful corpus for calculating the PMI-IR by issuing a query to a search engine to count the number of hits (as matching documents) which contain both words. The SO-PMI is based on the number of documents returned by a query to the search engine (hits). The SO-PMI is derived by aggregating the value of PMI-IR over a basket of canonical positive and negative words.

$$SO - PMI(word) = \log_2\left(\frac{\Pi_{pword \in Pwords} hits(wordNEARpword) \cdot \Pi_{nword \in Nwords} hits(nword)}{\Pi_{nword \in Nwords} hits(pword) \cdot \Pi_{nword \in Nwords} hits(wordNEARnword)}\right) \quad (3)$$

Since the numerator is a constant, it only needs to be calculated once for a given canonical set of positive words (Pwords) and negative words (Nwords). With the Google search engine, the `NEAR` operator can be implemented with the `*` operator. That is, the search `design * good` finds all documents in which the word *design* is separated by the word *good* by one or more words within a short context.

We selected a basket of 12 canonical positive and negative words. Adjectives and adverbs were selected based on most frequent occurrence in written and spoken English according to the British National Corpus [14, pp. 286-293]. Because this list is published separately, we joined both lists and ordered them by frequency per million words. We selected only those adjectives and adverbs which were judged positive or negative modifiers according to the General Inquirer corpus [http://www.wjh.harvard.edu/~inquirer/]. The basis for the selection of these frequently occurring words as the canonical words is the increased likelihood of finding documents which contain both the canonical word and the word for which the PMI–IR is being calculated. This increases the accuracy of the SO-PMI measurement. Table 1 lists the canonical adjectives their frequency per million words.

Table 1. Canonical positive and negative words for SO-PMI calculation

Positive Words	Negative Words
good (1276)	bad (264)
well (1119)	difficult (220)
great (635)	dark (104)
important (392)	cold (103)
able (304)	cheap (68)
clear (239)	dangerous (58)

In summary, the vector value for each word in a clause is calculated as the sum of the frequency of occurrence of the word multiplied by κ_2. The values for all words in a category are summed. The SO-PMI for adjectives and adverbs modifying a word are not altered by any correction factors; their SO-PMI values are summed and placed in the same category(ies) as the words that they modify.

Let us use the clause *These concepts are really interesting ideas* to demonstrate how to calculate the SO–PMI and form the 9–dimensional vector. Excluding the demonstrative pronoun "These", this clause contains 2 nouns {concepts (N,1), ideas (N,2)}, 1 verb {are (V,1)}, 1 adverb {really (A,1)} and 1 adjective {interesting (A,2)}. We exclude the verb "are" from further consideration since relational verbs for states of being (such as *to be*, *to have*, etc.) do not function to evoke any appraisal. Each of the vector values is calculated separately. For example, Pd_N is calculated by the formula $Pd_N = Pd_{N,1} \times SO-PMI_{N,1} + Pd_{N,2} \times SO-PMI_{N,2}$. We calculated the SO-PMI for these words by searching Google. The calculation yielded the following numerical results shown in Table 2. Note that the positive / negative values of the SO-PMI do not necessarily mean that a word is positive or negative in orientation.

Table 2. SO-PMI calculation for words in the sample clause

word	SO-PMI
concept	5.6
really	-12.1
interesting	-12.1
ideas	-12.3

The word *concept* appears in the WordNet logical group *noun.cognition*; the word *idea* appears in the logical groups *noun.cognition* and *noun.communication*. The logical group *noun.communication* (κ_1=1) contains words only in the category *Product*. The logical group *noun.cognition* (κ_1=1/2) contains words which are either in the category *Product* or *Process*; thus, the words *concept* and *ideas* could either refer to the product or the process of designing. That is, the system could construe this appraisal to be about the material embodiment of the concepts as interesting or that the process of coming to these concepts is interesting as in *These concepts **for designing** are really interesting ideas*. Here is a sample calculation of the correction factors κ_2. Note that $\kappa_{2,Pd}$(concept) = $\kappa_{2,Pr}$(concept).

$$\kappa_{2,Pd}(concepts) = \frac{1 \times \kappa_1(noun.cognition)}{1 \times \kappa_1^{-1}(noun.cognition)} = \frac{1}{4}$$

$$\kappa_{2,Pd}(ideas) = \frac{4 \times \kappa_1(noun.cognition) + 1 \times \kappa_1(noun.communication)}{4 \times \kappa_1^{-1}(noun.cognition) + 1 \times \kappa_1^{-1}(noun.communication)} = \frac{1}{3}$$

$$\kappa_{2,\Pr}(ideas) = \frac{4 \times \kappa_1(noun.cognition)}{4 \times \kappa_1^{-1}(noun.cognition) + 1 \times \kappa_1^{-1}(noun.communication)} = \frac{2}{9}$$

In summary,

$\kappa_2(concepts) = \left[\frac{1}{4}, \frac{1}{4}, 0\right]$ and $\kappa_2(ideas) = \left[\frac{1}{3}, \frac{2}{9}, 0\right]$.

Substituting the SO–PMI values:

$$Pd_N = \frac{1}{4} \times 5.6 + \frac{1}{3} \times -12.3 = -2.7 \qquad Pr_N = \frac{1}{4} \times 5.6 + \frac{2}{9} \times -12.3 = -1.3$$

The corresponding 9–dimensional vector is:

[-2.7, 0, -24.6, -1.3, 0, -24.6, 0, 0, 0]

The modifiers for a noun are placed in the respective categories. In this example, the word *ideas* is modified by *really interesting* to form the trigram *really interesting ideas*. Thus, the adjectives are placed in both the *Product* and *Process* dimensions in the vector.

IMPLEMENTATION AND RESULTS

To train the SVM classifier, on cohort of 3 native English speakers with a background in a design-related discipline (e.g., engineering, architecture, and computer-science) were tasked with reading and categorizing various design texts. The texts included formal and informal design text from various online sources and across various design-related disciplines. Each coder was paid to classify the texts. The rating cohorts were trained for one hour before they started formal rating work. The main purpose of training was to assure the cohorts could identify the proper category and its semantic orientation according to the context.

During coding, 2 of the 3 coders had to agree on the semantic meaning (category), semantic orientation (orientation), and the value of the orientation, that is, positive or negative. Working in two hour time blocks, the coders read various design texts, including formal design reports, reviews of designed works, reviews of designers, and transcripts of conversations of designers working together. Every thirty minutes, the coders took a "fatigue check" test to assess their performance. The fatigue test consisted of six appraisals that the second author had previous labelled. These constituted a set of "known" appraisals with correct content categorisation and semantic orientation. The appraisals were randomized so that the group would not receive two tests containing the same set of appraisals in the same order. The fatigue test usefully provides us a baseline for the "best" performance we could expect from the machine learning classifier as well as an assessment of the internal validity of the collected data. One-third of the data set was cross-categorised by the second author and a colleague coding per standard practice in protocol analysis to ensure agreement upon a reliable categorisation and sentiment classification. Finally, one of the researchers reviewed the cohorts' work to ensure that they were correctly following the rules for coding the categories and orientation according to the framework for the language of appraisal in design. (The coders will be asked to re-code text the author thought might be incorrectly labelled in the next phase of the research.) That is, we would ideally like to have the classifier perform at least as well as the human coders in categorising the text and its semantic orientation. In the practice of human labelling of data for training computational linguistic classifiers, the guideline of more than five votes per paragraph is used a baseline for confirming "correct" labelling of a text. This coding system satisfies this requirement and is reliable because at

least five people, three people from the student cohort and two researchers, agreed on the rating. The performance of the human coders is reported in Figure 3 for over 50 sessions of coding.

The performance of the coders with respect to accuracy of semantic categorisation (using the fatigue check test) of content was lower than we had anticipated, but tended to improve over time. The improvement is likely due to additional training provided to the coders and experience in understanding the grammatical patterns of appraisals. They were at times inconsistent, which is a measure of their fatigue. In coding the clause "*I was reeeally fussy with it*", which should be categorised in the Process category because the subordinate prepositional clause "with it" directs the appraisal to "it" via an anaphoric reference to a doing, the coders categorised this clause as being about Process 20 of 24 times and as being about People 4 of 24 times. Their performance associated with semantic orientation was much more consistent, with better than 90% accuracy over most of the sessions. Using the previous example clause again, they correctly identified this clause as being negative, as fussiness (at least in Australia) is considered negative. Only once did the cohorts think that fussiness was a positive characteristic. Pang [12] found that human-based classifiers were accurate only 58% to 64%; thus, the performance of our coders is consistent with other studies and is likely to be the "best" that could be expected. As a "rule of thumb," the coder's performance data suggests a "gold standard" of 80% for semantic categorisation and 90% accuracy for sentiment classification.

In total, 1500 paragraphs will be collected, 900 to serve as the training set and 600 as the validation set. To date, we have collected more than 150 paragraphs (A paragraph is at minimum 50 characters.) for each category, which is sufficient for training and validation purposes. The size of the training set is sufficient if the accuracy of the results on the validation set is higher than an expected minimum. Generally, it is been comparatively to locate positive evaluative design text rather than negative design text, in particular in the evaluation of designers (i.e., appraisals of people).

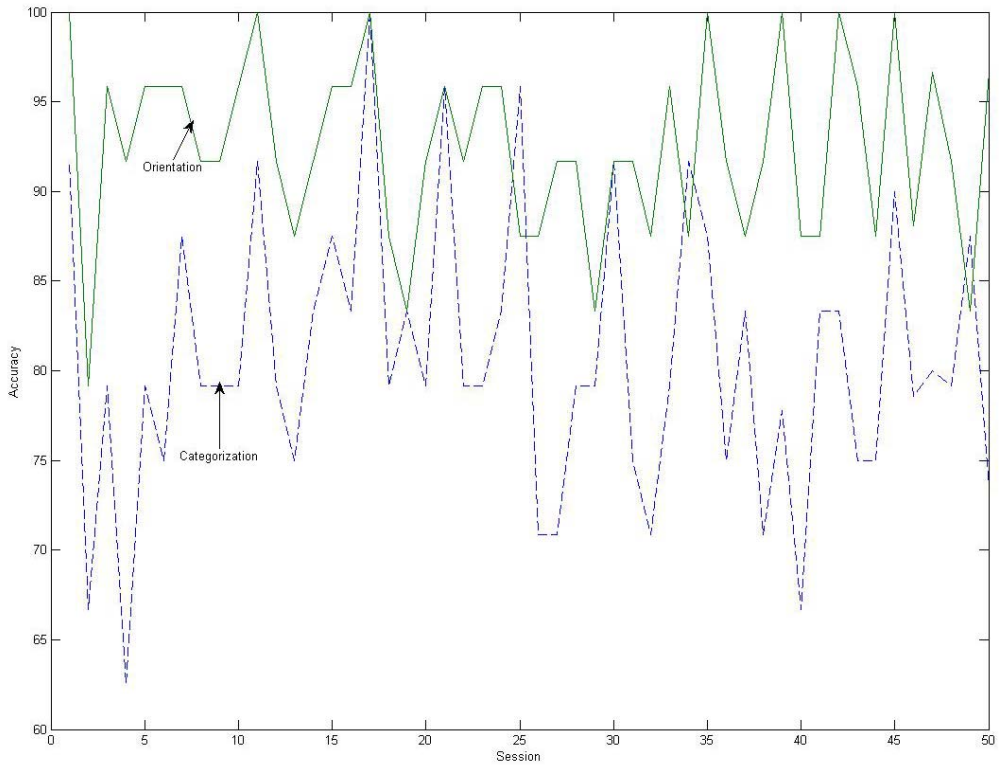

Figure 3. Performance of human coders

To speed up the performance of the system, we cached the SO-PMI values of all adjectives and adverbs in the WordNet dictionaries, about 26,000 words. Originally, we used Google's SOAP API; however, Google limits the SOAP-based searches to 1,000 queries per day for each search key. As such, we wrote a script in PERL using LWP::UserAgent to query Google directly and "screen scrape" the number of hits. Since the estimated number of hits varies depending upon which Google server(s) are actually searched, we ran the queries ten times and record the average value. Through

experimentation, we found that spitting the word lists into groups of 20 and running them in parallel provided the best performance.

Table 3 summarizes the performance of the SVM-based category and sentiment classifier on this initial set of collected data. The numbers reported are the average performance of each classifier over 10 runs followed by the range of the worst and best performance. In categorisation ($SVM_{category}$), the classifier could perform at best as well as the lower bound of performance by the coders. In semantic orientation ($SVM_{orientation}$), however, the classifier did not perform as well as the human coders. In only three cases, PR(Neg) and PP(Neg) and PP(Pos) did the orientation classification approach the worst performance of the human coders. Nonetheless, the performance of $SVM_{orientation}$ is better than the random choice baseline of 50% for semantic orientation; likewise the performance of $SVM_{category}$ is better than the random choice baseline of 33% for categorisation.

Table 3. Performance of sentiment classifier

Data	#Original	#Training	#Validation	$SVM_{category}$	$SVM_{orientation}$
PD(Neg)	136	68	68	61.76±4.41%	67.65±5.89%
PR(Neg)	88	44	44	59.09±6.82%	75±9.09%
PP(Neg)	73	37	36	63.89±11.11%	73.61±9.72%
PD(Pos)	322	161	161	61.49±2.48%	65.84±4.35%
PR(Pos)	277	139	138	62.68±6.16%	66.30±3.99%
PP(Pos)	113	57	56	61.61±9.82%	71.43±12.5%

The best performance of our sentiment classifier is very close to the close to the performance of SVM-classifiers using relatively "clean" text of movie reviews as reported by Pang [12]. Our $SVM_{orientation}$ classier is nearing the accuracy of Pang's SVM classifier using unigrams (72.8% accuracy) and adjectives (75.8% accuracy). We should note that Pang's classifier worked on multi-paragraph texts rather than short collections of clauses, which is necessary to capture the dynamic changes in appraisals in design texts. Thus, the comparison between the two systems is not entirely homologous. It is not likely that Pang's system could operate over the types of text segments our system works on.

In particular, our system had difficulty with design text from design conversations which contained some grammatical mistakes and connected the verbalizations of several designers into a single paragraph. These types of texts though are representative of the "dirty" text one would encounter in design texts. For example, the system incorrectly categorised the semantic content and semantic orientation of the following paragraph (Category: Product; Orientation: negative)

if you make it really flat it is not so good that's not really small.
This doesn't belong to the drawing up there. It doesn't have to be roand.
you can't see then, because they are infinitesimal small.
I didn't understand that with the cantilever that is only the principle how you lamp must look like

What is most interesting, however, are the cases in which the $SVM_{category}$ classifer correctly identified the category but the human coders were incorrect. For example, the coders classified the following paragraph in the category People with negative orientation. It should have been Process with negative orientation (as judged by the second author).

I spent my time paring my work down to the essence, to the bones. I spent my time reducing everything to Frutiger and to line and vector and plane. I can't help it. This is not to say I am not an angry person. I'm tired of the narrow language, the small sandbox, the limits of what we deem "good design"

This paragraph should have been in the category Process (The writer is appraising the actions of "paring down" the work and the normative design process, named as "narrow language," which constrains the designer's choices.) which is what the $SVM_{category}$ classifer correctly identified. Thus, the machine learning classifier can, in some instances, be more accurate than the human coders. This same finding has been found in other sentiment analysis research.

CONCLUSION

In this paper, we presented a computational model for the language of appraisal in design. Based on the computational model, we trained and validated a support vector machine based system to categorise and classify the text according to semantic orientation. Using a compact vector-based representation of design text, the classifier was able to perform at least as well as the human coders and is approaching the performance of other SVM classifiers which were operating on clean and rather uniformly written text. More work is necessary to improve the accuracy of the human coders in tagging the training data and handling poorly formatted text. The accuracy of the SVM classifiers may improve as we generate more training data so that there is not an imbalance in the number of positive and negative text and the number of text in each category.

We are continuing to collect more data for the appraisals and working to improve the quality of the labelled data. We are currently comparing our method with Pang's methods on our data set and our method on Pang's data set and experimenting with the effect of the correction factors.

The eventual aim of the research is to develop a system which can process large amounts of design text to extract the sentiments contained therein. To do this, we will divide texts into coherent multi-sentence discourse units which contain different sub-topics [15]. If the words in the text unit contain sufficient information for SVM classifiers, then the system proceeds to classify the category and sentiment. If not, the next unit is added until there is sufficient context. We can then investigate the frequency and types of appraisals used and examine the patterns of appraisals. For example, we could address whether designers appraise process first and then product toward the end of the designing. Second, by mapping attitude in design text to patterns of design thinking, we might also discover the co-existence of affective processing and rational cognitive processing given the hypothesized links between language and the human affective system. Investigations of words that describe attitudes may shed light on the way that affectively-derived attitudinal stances affect, influence, or direct design cognition. A system that can understand the deployment of designers' attitudes in design text may also be usefully applied to design rationale explanation systems and case-based information retrieval systems. They may also be used to monitor communication between participants in a group design situation to detect potential design problems or breakdowns in shared understanding. As such, we believe that understanding the language of appraisal in design is a key enabling technology for a wide range of design support systems.

ACKNOWLEDGEMENTS

This research is supported by an Australian Research Council grant DP0557346 and an Australian Postgraduate Award (APA) for the first author.

REFERENCES

[1] Cross, N., Christiaans, H. and Dorst, K., eds. *Analysing Design Activity*. (John Wiley & Sons Ltd, Chichester, 1996).

[2] Dong, A. How am I doing? The language of appraisal in design. In Gero, J.S., ed. *Design Computing and Cognition '06 (DCC06)*, pp. 385-404 (Kluwer, Dordrecht, 2006).

[3] Leech, G.N. *Principles of pragmatics*. (Longman, London, 1983).

[4] Sperber, D. and Wilson, D. *Relevance : communication and cognition*. (Blackwell Publishers, Cambridge, MA, 1995).

[5] Martin, J.R. Beyond Exchange: APPRAISAL Systems in English. In Hunston, S. and Thompson, G., eds. *Evaluation in Text: Authorial Stance and the Construction of Discourse*, pp. 142-175 (Oxford University Press, Oxford, 2000).

[6] Nussbaum, M.C. *Upheavals of thought: the intelligence of emotions*. (Cambridge University Press, Cambridge, 2001).

[7] Ortony, A., Clore, G.L. and Foss, M.A. The referential structure of the affective lexicon. *Cognitive Science*, 1987, 11(3), 341-364.

[8] Wiebe, J.M., Bruce, R.F. and O'Hara, T.P. Development and use of a gold-standard data set for subjectivity classifications. *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pp. 246-253 (Association for Computational Linguistics, Morristown, NJ, USA, 1999).

[9] Hatzivassiloglou, V. and McKeown, K.R. Predicting the semantic orientation of adjectives. *Proceedings of the eighth conference on European chapter of the Association for Computational*

Linguistics, pp. 174-181 (Association for Computational Linguistics, Morristown, NJ, USA, 1997).
[10] Turney, P.D. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 417-424 (Association for Computational Linguistics, 2001).
[11] Sun, A., Lim, E.-P., Benatallah, B. and Hassan, M. FISA: Feature-Based Instance Selection for Imbalanced Text Classification. In Ng, W.K., Kitsuregawa, M., Li, J. and Chang, K., eds. *Advances in Knowledge Discovery and Data Mining, 10th Pacific-Asia Conference, PAKDD 2006*, pp. 250-254 (Springer Berlin, Berlin, 2006).
[12] Pang, B., Lee, L. and Vaithyanathan, S. Thumbs up? Sentiment Classification using Machine Learning Techniques. *2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 79-86 (Association for Computational Linguistics, University of Pennsylvania, Philadelphia, PA, USA, 2002).
[13] Joachims, T. *Learning to classify text using support vector machines*. (Kluwer Academic Publishers, Boston, 2002).
[14] Leech, G., Rayson, P. and Wilson, A. *Word Frequencies in Written and Spoken English based on the British National Corpus*. (Pearson Education Limited, Harlow, UK, 2001).
[15] Hearst, M.A. Multi-paragraph segmentation of expository text. *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pp. 9-16 (Association for Computational Linguistics, Morristown, NJ, USA, 1994).

Contact: Dr Andy Dong
Key Centre of Design Computing and Cognition
University of Sydney
Wilkinson Building (G04)
Sydney 2006 Australia
Phone: +61 (02) 9351 4766
Fax: +61 (02) 9351 3031
e-mail: andy.dong@usyd.edu.au
URL: http://www.arch.usyd.edu.au/~adong/