A Probabilistic Representation of Systemic Functional Grammar

Robert Munro
Endangered Languages Archive
Department of Linguistics
School of Oriental and African Studies
University of London
rmunro@soas.ac.uk

Abstract

The notion of language as probabilistic is well known within Systemic Functional Linguistics. Aspects of language are discussed as meaningful tendencies, not as deterministic rules. In past computational representations of functional grammars, this probabilistic property has typically been omitted. This paper will present the results of a recent project aimed at the computational learning, representation and application of a fundamentally probabilistic functional grammar. Recent advances in machine learning have allowed the large scale inference of truly probabilistic representations of language for the first time. In this work, a machine learning algorithm is developed that learns aspects of a functional grammar from labeled text. This is represented probabilistically, in the sense that there is a measurable gradation of functional realisation between all categories. Looking at a single term, this allows that term to be described as realising multiple functions simultaneously. Looking at all the terms in a text or register, this allows us to examine the relationships between the functions with respect to the closeness and/or overlap of functions, and the extent to which these relationships differ between different texts or registers. With a focus on function within the noun phrase (nominal group), the methodology is shown to infer an accurate description of functional categories that classifies new examples with above 90% accuracy, even across registers of text that are very different from the text that was learned on. Importantly, the learner is deliberately restricted from remembering specific words, so that the functions are (necessarily) learned and represented in terms of features such as part-of-speech, context and collocational tendencies. This restriction allows the successful application to different registers and demonstrates that function is much more a product of context than a property of the words themselves. The inferred grammar is also shown to have interesting applications in the analysis of layers of delicacy. The discovery of finer delicacies occurs with a high level of sophistication, indicating a potential for the automated discovery and representation of lexis as most delicate grammar.

1 Introduction

Research describing functional grammars is often prefaced with strong assertions that the grammars (and therefore the systems, constraints, constituencies and dependencies) are probabilistic, with aspects of language variously described as a gradational, fuzzy and/or cline (Hasan, 1987; Halliday, 1994; Tucker, 1998; Fawcett, 2000; Halliday, 2002). While functional categories have long been described as meaningful tendencies in a continuous space, these shades of grammar have rarely been explored.

More commonly 'probabilistic linguistics' is used to refer to confidences across multiple deterministic models, or within a single deterministic model (probability of constituency) rather than a single gradational model. This is largely because probabilistic parsing techniques have grown out of deterministic theories.

The functions of modification within the noun phrase (nominal group) provide good examples for describing such gradations. Classifiers such as those in 'the *1,000* metre race' and 'the *red* wine' still function close to the Numerative and Epithet from which they originated, and will typically realise both functions. Gradient representations of function are necessary to describe this gradience of realisation.

Even where individual instances of functional modification are not gradationally realised, gradational modelling is still necessary. Common solutions for describing some new object/concept include creating a new word (often through compounding), creating a new sense for an existing word or using multiple words. Combinations of the three are possible, as can be seen in the phrase 'notebook computer'. 'Notebook' was created as a compound, 'notebook computer' became a multi-word entity and now 'notebook' alone has the new sense of a type of computer. There is little ambiguity between Epithets, Classifiers and Things here, but given that the uptake of the new term/sense will not be uniform and that a given person's use may not be consistent (they may only use the new sense of 'notebook' in the context of computers). This shows that the computational modelling of nominals still needs to be gradational in modelling across deterministic instances.

It might be assumed that part-of-speech is a good indicator of functional modification, giving an insight into the part of the world we represent in a noun phrase (the experiential metafunction of nominals), with exceptions being rare or idiosyncratic. Previous functional parsers have relied on this assumption. In this work, it is demonstrated that assuming the unmarked functions given by part-of-speech and word order will only account for about half the instances of Classifiers in the registers investigated here, showing that more sophisticated modelling is required for computational representations.

The difficulty in building a fundamentally probabilistic model of a grammar lies in defining the gradations. Defining a probability distribution across two or more categories in terms of a large number of features is a difficult manual task, and it is not surprising that previous models have relied on computational processing over labelled data to calculate these. Machine learning is the most popular method for combining this with the ability to predict new instances. In this work, a new machine learning algorithm, *Seneschal*, is developed that models tendencies in the data as an optimal number of soft clusters, using the probability of membership of a cluster to make supervised classifications of new data.

The most sophisticated models utilising machine learning have been probabilistic-context-free grammars and stochastic grammars that have focused their interpretation of results on the accuracy of the inferred syntax (Bod, 1993; Collins, 1999; Charniak, 2000; Johnson, 2003). In a functional lexicogrammar this roughly corresponds to only the logical metafunction (although the feature spaces used are much richer, and gradational models have been suggested (Aarts, 2004; Manning, 2003)) but similar techniques can be used for modelling more complicated functional relationships.

In Systemic Functional Grammar (SFG), computational representations and applications of artificial intelligence are not new, but most work in this area has focussed on language generation (Mann and Mattheissen, 1985; Matthiessen and Bateman, 1991) and machine learning has not previously been used

in the inference of a functional grammar.[1] The most well-known systemic parser is *WAG* (O'Donnell, 1994). It was the first parser to implement a full SFG formalism and it performed both parsing and text generation. Drawing from work with context free grammars, it treated the grammar as deterministic, giving good but limited coverage. It didn't attempt the disambiguation of the unmarked cases of the functions of words. There have been a number of earlier implementations of SFG parsers, but with more limited coverage, (Kasper, 1988; O'Donoghue, 1991; Dik, 1992). For German, Bohnet, Klatt and Wanner implemented a successful method for the identification of Deictics, Numeratives, Epithets and Classifiers within the noun phrase by implementing a bootstrapping algorithm that relied on the general ordering of the functions (Bohnet et al., 2002). They were able to assign a function to 95% of words, with a little under 85% precision. A more extensive review of related work can be found in Munro (2003b).

2 Machine Learning for Linguistic Analysis

Supervised machine learning algorithms are typically used as black boxes, restricted to classifying independent categories or flat structures (for an exception in computational linguistics see (Lane and Henderson, 2001)). Unsupervised machine learning is a technique for finding meaningful rules, clusters and/or trends in unlabeled data and are more commonly used to discover fuzzy (soft), hierarchical and/or connectionist structures. As such, the goal of unsupervised learning is often analysis, not classification.

In this work, unsupervised and supervised learning are combined so that a single model can be described in both its ability to identify functions and to provide information for detailed analysis.

Here, we seek to discover finer layers of delicacy by looking for meaningful clusters within each function. In SFG 'delicacy' describes the granularity chosen in describing a given function. For example, in Table 1, the terms 'one' and 'first' both function as Numeratives, but could have been broken down into the more delicate functions of Quantitatives and Ordinatives respectively. As more delicate functions are sought, more constraints and tendencies can be described, and therefore we can build a more informative model.

3 Scope of Study

This study explored functional categories across all groups/phrases of English, but only those of the noun phrase are described here. See Munro (2003b) for the results and analysis of the other functions. Examples of nominal functions taken from the corpus used here are given in Table 1.

Definitions are drawn from Halliday (1994), Matthiessen (1995) and O'Donnell (1998). Below we describe the functions that are the target of the supervised classification (in **bold**), and those that were/could be discovered through unsupervised learning at finer layers of delicacy (in *italics*):

Deictic: Deictics fix the noun phrase in relation to the speech exchange, usually through the orientation of the speaker. At a finer layer of delicacy this includes *Demonstratives*, ('this', 'that', 'those'), and *Possessives*, ('my', 'their', 'Dr Smith's').

Ordinative: An Ordering Numerative, ('first', '2nd', 'last').

Quantitative: A Quantitative Numerative, ('one', '2', 'many', 'few', 'more'). They may used *Discursively*, ('the *12* championships') or simply be *Tabulated* results, which was common here due to the choice of registers.

[1] Machine learning has been used to learn formal grammars that include functional constraints such as Lexical Functional Grammar (Bresnan, 2001), a theory that is also still evolving. Its F-structure *could* be described as a functional grammar by some (or arguably many) definitions. Describing the relationship between LFG and SFG theories is outside the scope of this paper, but it is a comparison that is probably overdue.

Deictic	Numerative	Epithet	Classifier	Thing
the	third	fastest		time
the				Atlanta Olympics
Burundi's			5,000 metres	champion
their	first			World Cup
Colombia's		former	team	boss
the			defending	champion
a		controversial		final
the		Superman	riding	style
	three first-round			matches
the			bronze	medal
		real	data	sets
		robust	parametric	methods
a	single		microarray	chip
the		bootstrapped		version
her		own		fortunes
the		smooth unmarked		outline
a		little	parchment	volume
this	one			scene

Table 1: Example of functional categories

Epithet: Describes some quality or process. At a finer layer of delicacy there are *Attitudinal* Epithets, ('the *ugly* lamp'), and *Experiential* Epithets, ('the *red* lamp'). They are most commonly realised by an adjective, but are also commonly realised by a verb, ('the *running* water').

Classifier: Describes a sub-classification. Classifiers are commonly realised by a noun, ('the *table* lamp'), a verb, ('the *running* shoe'), or an adjective, ('the *red* wine'), but other realisations are also possible. Classifiers are commonly thought of as providing a taxonomic function, a *Hyponymic* Classifier. They may also be used to expand the description of the Head: an *Expansive* Classifier (Matthiessen, 1995). The latter are classifications that can more easily be reworded as Qualifiers or expanded clauses, for example, 'knee surgery' can be re-written as 'surgery of the knee'. In the work described here, they were a particularly interesting cases, as they allowed anaphoric reference of non-Head terms, ('she underwent *knee* surgery after *it* was injured...').

Thing: Typically the semantic head of the phrase. Some entity, be it physical, ('the *lamp*'), or abstract, ('the *idea*'), undergoing modification by the other noun phrase constituents. Delicacies within Thing include into *Countable* and *non-Countable*, *Named Entities* (*First*, *Intermediate* and *Last* Names), and those simply realised by nouns and non-nouns. Of all the functions in the noun phrase, variation in function of the Thing corresponds most strongly with variation in the function of the phrase such as the Referring and Informing functions of a noun phrase (the heads of such phrases are called *Stated* and *Described* Things respectively). When a noun phrase is realised by a single word, the function is best described in terms of the function of phrase.

4 Testing Framework

4.1 Algorithm

Seneschal is a hybrid of supervised and unsupervised clustering techniques. It has been demonstrated to be generally suited to the efficient supervised classification and analysis of various data sets (Munro,

2003a). Similar to the EM algorithm and Bayesian learning, it seeks to describe the data in terms of an Information Measure (IM), combining agglomerative and hierarchical clustering methods.

Given an item i with value i_α for categorical attribute α, and given that i_α occurs in cluster C with frequency $f(i_\alpha, C)$, within the data set T, i's information measure for n categorical attributes for C with size $s(C)$ is given by:

$$IM(i, C) = \sum_{\alpha=1}^{n} -ln \frac{f(i_\alpha, C) + 1}{s(C) + (1 - \frac{f(i_\alpha, T)}{f(i_\alpha, T) - s(T)})} \quad (1)$$

Given an item i with value i_β for continuous attribute β, i's information measure for n continuous attributes for a cluster C that for attribute β that has an average of $\mu_C\beta$ and standard deviation of $\sigma_C\beta$ is given by:

$$IM(i, C) = \sum_{\beta=1}^{n} \frac{(i_\beta - \mu_C\beta)^2}{2\sigma_C\beta^2} \quad (2)$$

The algorithm maps to an SFG in the following ways:

1. It is probabilistic, giving a gradation of membership across all categories.

2. The algorithm treats all classes independently. If the feature space describes two classes as overlapping, this will be apparent in the model, capturing the overlapping categories. This is particularly important here, as we need a learner that represents each class as accurately as possible. A learner that only represents categories by defining boundaries between them goes against our knowledge of multiple and gradational realisation.[2]

3. The discovery of the optimal number of clusters within a class maps to the task of describing the emergent finer layers of delicacy within a function.

4. Beyond a minimum threshold, the algorithm is *not* frequency sensitive, so it will not intrinsically favour the patterns of realisation of functions in the training corpus. This makes it more appropriate than other algorithms that seek to discover an optimal number of clusters by strong *a priori* assumptions of optimal cluster size.[3]

4.2 Corpora

One training corpus and four test corpora were used. The process of manually tagging the corpus with the correct functions took about 20 hours, performed by two linguists with input from domain experts in the fields of bio-informatics and motor sports. Here, we simply labelled a term with its most dominant function.[4]

The training corpus comprised 10,000 words of Reuters sports newswires from 1996. It was chosen because Reuters is one of the most common sources of text used in Computational Linguistics, and the choice of only sports newswires was motivated by two factors: taking the corpus from only one register

[2] In terms of Aart's definitions of Subsective and Intersective Gradience (2004), the probability of cluster membership described here is Subsective Gradience, and the cross-cluster costs are Intersective Gradience. Note that if the clusters were not formed independently and prevented from overlapping, then the probability of membership could not be thought of as Subsective Gradience as the cluster (category) would be partially defined in terms of its intersection with other categories.

[3] In the work reported here, assuming that the relative frequencies of the categories are the same in the test set is the equivalent to the learner assuming that all text is sports newswires. This is a well-known problem in natural language processing, known as domain dependence, and the algorithm described goes some way in addressing the problems. Gradation not wholly dependent on observed frequency is, in itself, a desirable quality when dealing with sparse data.

[4] It would be interesting to see how explicitly defining gradient membership for the training data would affect the model learned, but this would be a complicated task in a largely untested area of machine-learning.

was desirable for testing purposes, and sports terminology is known to be an interesting and difficult one to study as, for example, it is necessary to learn that a 'test match' is a type of cricket match and a '1,000 metre race' is a type of race (this is what allows 'I won *the 1,000*' and 'They played *the test*' to be grammatical).

Four testing corpora were used, all of approximately 1,000 words. The register (domain) dependence of NLP tasks is well known so they were drawn from a variety of registers:

1. Reuters sports newswires from 1996 (Reuters-A), from the same corpus as the training set.

2. Reuters sports newswires from 2003 (Reuters-B). This is presumed to be the same register, but is included to test the extent to which 'topic shift' is overcome.

3. Bio-Informatics abstracts (BIO-INF), to test the domain dependence of results in a register with a high frequency of rare words/phrases, and with some very large and marked Classifier constructions.

4. An excerpt from a modernist fiction (MOD-FIC), 'The Voyage Out', Virginia Woolf (1915), to test the domain dependence of results on an Epithet frequent register.

4.3 Features

part-of-speech : POS was assign *mxpost* (Ratnaparkhi, 1996). It was modelled to a context window of two words. The standard codes for POS are used here.

POS augmentations : Features representing capitalisation and type of number were used, as *mxpost* over assigned NNP's to capitalised words, and under-assigned numbers. Number Codes: NUM = only numerals, WRD = word equivalent of a numeral, MIX = a mix, eg '6-Jan', '13th'.

punctuation : Features were included that represented punctuation occuring before and after the term. Punctuation itself was not treated as a token.

collocational tendencies : Features were included that represented the collocational tendencies of a term with the previous and following words and the ratio between them. These were obtained automatically using the *alltheweb* search engine, as it reports the number of web documents containing a searched term, and could therefore be used to automatically extract measures from a large source. For two terms 'A' and 'B', this is given by the number of documents containing both 'A' and 'B', divided by the number of documents containing the bi-gram 'A B'.

repetition : (self-co-occurrence) The observed percentage of documents containing a term that contained more than one instance of that term. These were taken from a large corpus of about one hundred thousand documents of Reuters newswires, Bio-Informatics abstracts, and the full 'The Voyage out' split into equivalent sized chunks.

phrase context and boundary : The following and previous phrase types were included , as was the term's position in its own phrase.

The words themselves were omitted from the study to demonstrate that functions are not simply a property of a word (like most parts-of-speech) but a product of context. It is expected that allowing the algorithm to learn that a certain word has previously had a certain function would give a small increases in accuracy but a substantial increase in domain dependency.

Other additional features were considered, such as the use of lexico-semantic ontologies and more complex modelling of repetition, but were not included here to simplify the analysis (or were investigated independently).

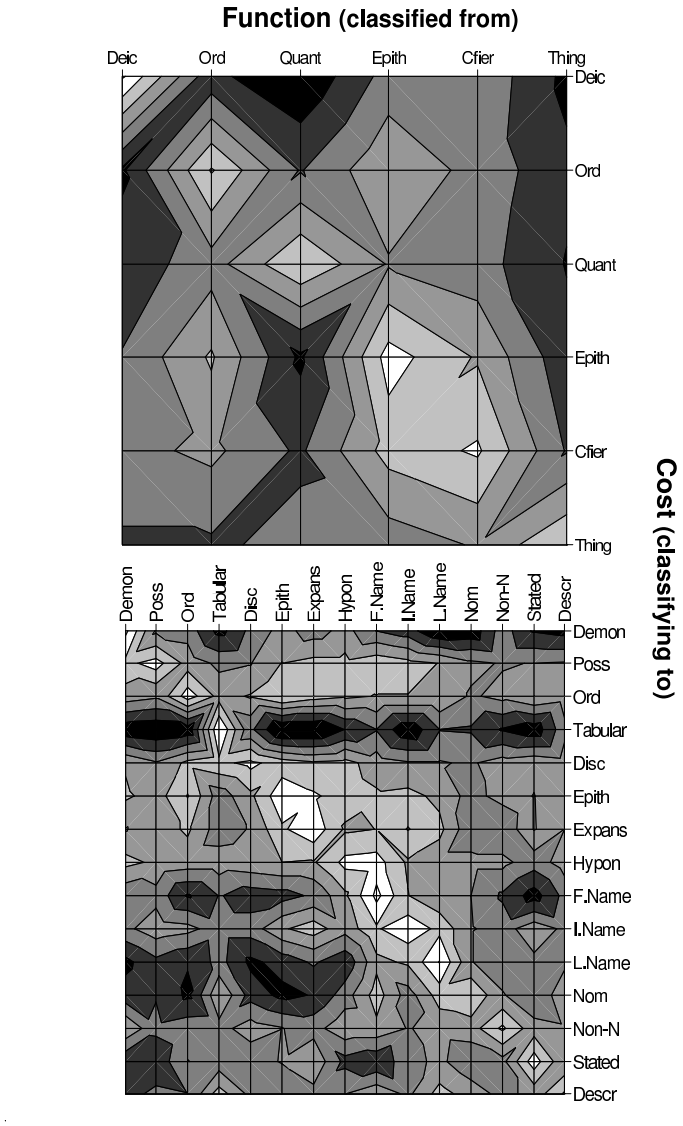

Figure 1: Gradational realisation: the IM costs between functions, at two layers of delicacy

5 Analysis

The raw accuracy of classifying functions within the noun phrase was 89.9%. The accuracy of a parser only seeking to describe unmarked functions based on part-of-speech and word order would classify with 82.6% accuracy on these test corpora, so the method here almost halved the error of existing methods. This baseline was reached by *Seneschal* after only 5% of the data was seen (the overall accuracy for all other group/phrase types was over 95%).

A confusion matrix (number of cross-categorical errors) doesn't capture the probabilistic nature of the distribution. Here, the gradations are measured as the average IM cost for assigning items between all clusters/functions. Figure 1 represents the pairwise calculations of gradations topographically. The top map shows the relationships between the targets of the supervised task, the bottom map between more delicate clusters/functions. If there were no probabilistic boundaries between the functions, the maps in would be a diagonal series of white peaks on a black background with the height of a peak

Function	Significant Features	Examples
Demonstrative	*pos:* DT=80%, PRP=15% *prev phrs:* prep=56%, verb=36% *next phrs:* prep=44%, noun=23%	'a', 'the', 'these'
Possessive	*pos:* DT=32%, POS=25%, NNP=24% *prev phrs:* noun=48%, prep=39% *next phrs:* verb=57%, prep=32%	'our', 'The Country Club's'
Tabular (Quantitative)	*num type:* NUM=69%, MIX=18%, *prev phrs:* noun=100% *phrs end:* yes=92% *coll prev:* (ave= 0.10, var= 0.04) *coll next:* (ave= 0.05, var= 0.01)	'1, 2, 20'
Discursive (Quantitative)	*num type:* WRD=39%, MIX=26% *prev phrs:* prep=40%, verb=30% *phrs end:* yes=41% *coll prev:* (ave= 0.02, var= 0.00) *coll next:* (ave= 0.11, var= 0.06)	'*2* cars,' 'the *twelve* championships,'
Ordinative	*num type:* ORD=88%, WRD=8% *prev phrs:* prep=42%, verb=36% *phrs end:* yes=23% *coll prev:* (ave= 0.24, var= 0.09) *coll next:* (ave= 0.22, var= 0.15)	'the *third* fastest', 'the *top* four'

Table 2: Properties of the Deictic and Numerative functions

representing how tightly that function was defined by the features. In this study, the significance of discovered delicacies is precisely the difference in the complexity of the two maps in Figure 1.

The ordering of the functions in Figure 1 is simply the general observed ordering. What is not represented in Figure 1 is the attributes that were the most significant in distinguishing the various functions, that is, the attributes that contributed most significantly to a given 'valley'.

The remainder of this section describes the more delicate functions, including the features that were the most significant in distinguishing them. It is important to remember that these features are *both* a description of that function and the reason that *Seneschal* identified them, and that co-significant features are also features that correlated with each other for that function.

5.1 Deictics and Numeratives

There were two clusters/functions discovered within the Deictic function corresponding well to the more delicately described functions of Demonstratives and Possessives. The Possessive cluster contained mostly Genitives in the form of embedded noun phrases. The profiles of these and the Numeratives are given in Table 2. While differentiation in the part-of-speech distributions are as expected, the phrase context is particularly interesting, as it shows that the Possessives are more likely to occur in the Subject position, given by their being more likely to occur before a verb phrase and after a noun phrase.

The Quantitative function was divided into two sub-clusters, here simply labelled 'Tabular' and 'Discursive' as they are divided along the lines of reported results and modifiers within a phrase. As Figure 1 shows, the relationship between the tabulated numbers and the other functions is the least probabilistic. It would be easy to assume that no relationship existed between them at all, but they do leak into each other in the final phrase in sentences like 'Fernando Gonzalez beat American Brian Vahaly *7-5, 6-2*'.

The Ordinatives differentiate themselves from the Quantitatives by possessing particularly strong collocational tendencies with the previous words, as an Ordinative is much more likely to require exact determination from a small selection of closed-group words.

Function	Significant Features	Examples
Epithet	*pos:* JJ=78%, RB=4%, JJR=4% *prev pos:* DT= 47%, IN= 10% *repetitn:* (ave= 0.26, var= 0.04) *coll prev:* (ave= 0.16, var= 0.05) *coll next:* (ave= 0.20, var= 0.13)	'*erratic* play', '*bigger* chance'
Expansive (Classifier)	*pos:* JJ=34%, NN=31%, NNP=16% *prev pos:* IN= 30%, NN= 16% *repetitn:* (ave= 0.42, var= 0.06) *coll prev:* (ave= 0.02, var= 0.00) *coll next:* (ave= 0.34, var= 0.19)	'*knee* surgery', '*optimization* problems'
Hyponymic (Classifier)	*pos:* NN=53%, JJ=17%, NNP=14% *prev pos:* JJ=37%, DT=27% *repetition:* (ave= 0.47, var= 0.04) *coll prev:* (ave= 0.26, var= 0.15) *coll next:* (ave= 0.30, var= 0.16)	'the *gold* medal', 'the *world 3,000 metres* record'

Table 3: Properties of the Epithet and Classifier functions

5.2 Epithets and Classifiers

The difference between Attitudinal and Experiential Epithets is probably the most common example of delicacy given in the literature. Nonetheless, either the attributes failed to capture this, the learner failed to find it or it wasn't present in the corpora, as this distinction was not discovered.

The profiles for Epithets and Classifiers are in Table 3. Within Classifiers, the clusters describe Classifiers that corresponded well to the functions of Expansive and Hyponymic Classification.

Expansive Classifiers are more closely related to Epithets, and Hyponymic Classifiers more closely related to multi-word Things, so the distinction is roughly along the lines of marked and unmarked Classifiers, although both contain a considerable percentage of marked cases realised by adjectives. It is interesting that Figure 1 shows that the difference between the types of Classifiers is one of the most well defined indicating that the adjectives realising marked Hyponymic Classifiers were confidently identified.

Hyponymic Classifiers are much more likely to occur in compound or recursive Classifying structures (Matthiessen, 1995), which is why they exhibit strong collocational tendencies with the previous word, while the Expansive Classifiers exhibit almost none. As expected, the collocational tendencies with the following word was greater for Classifiers than for Epithets, although the variance is also quite high.

The selection of parts-of-speech context also differs between functions. While the Hyponymic Classifiers seem to follow adjectives, and therefore are likely to follow other Classifiers or Epithets, the Expansive Classifiers most commonly follow a preposition, indicating that they are likely to occur without a Deictic or Numerative and without sub-modification.

Epithets generally occur more frequently than Classifiers, so the probability of repetition of a Classifier within a document being almost twice as high is especially significant.

5.3 Thing

The clusters that were discovered can be roughly divided between those describing Named Entities (First, Intermediate and Last Names), those with the phrase realised by a single word (corresponding to Nominative and non-Nominative functions within the clause) and nominals corresponding to the Referring and Informing functions of a noun phrase. The properties of the Named Entity and Nominative/non-Nominative functions are well-known and there were few surprises in the features describing them here.

Here, we investigate the relative frequencies of functional modification of Stated and Described Things, assumming that most are some combination of Referring and Informing functions (O'Donnell,

Function	Significant Features	Examples
Stated (Thing)	*phrs start:* yes=2% *pos:* NN= 67%, NNS=30% *prev pos:* JJ= 32%, DT= 27%, NN= 16% *prev phrs:* prep=46%, verb=33%, noun=12% *next phrs:* prep=45%, noun=21%, verb=10% *coll prev:* (ave= 0.31, var= 0.16) *coll next:* (ave= 0.08, var= 0.02)	'media *questions*', 'the *invitation*', 'such *comparisons*'
Described (Thing)	*phrs start:* yes=58% *pos:* NN= 45%, NNS=13% *prev pos:* J= 21%, NN= 21%, NNP= 19% *prev phrs:* noun=78%, verb=12%, prep=8% *next phrs:* noun=91%, verb=4%, conj=2% *coll prev:* (ave= 0.10, var= 0.05) *coll next:* (ave= 0.01, var= 0.00)	'20.67 *seconds*', 'former *winner*', 'our *implementation*'

Table 4: Properties of the Stated/Described Things

1998). The distinction between the two may be seen in the choices made within the Deictic and Classification systems of delicacy. While the Stated Thing is twice as likely to be modified by a Deictic, over 80% of these are Demonstratives, which don't feature in the Described's modifications. This trend is reversed for Classifiers. The Described Things are more than twice as likely to be modified by a Classifier, and within this over 70% of cases are Expansive, as opposed to about 25% for the Stated Things.

As Figure 1 shows, the trend of Hyponymic Classifiers being more closely related to the Thing is *reversed* for the Stated Things: unlike other Things, a Stated Thing is most closely related to an Expansive Classifier. An explanation for this reversal is that it represents that a Hyponymic Classifier may itself undergo Classification while an Expansive Classifier generally does not, although the Stated thing seems to define a number of aberrant 'hills and valleys' with the intersection of the other functions in Figure 1, indicating it may represent something more complicated.

Not described in Figure 2 is that the percent of Epithets is much less than the percentage of preceding adjectives given in Table 4, indicating that markedness is common to both. The fact that the Stated is twice as likely as the Described to be modified Epithetically indicates that the labels given to them are not quite sufficient in describing the complexities of the differences. This also demonstrates that at finer layers of delicacy, the variation in function can quickly become very emergent, even when the corresponding parts-of-speech and other surface-level phenomena independently differ only slightly.

5.4 Inference of unmarked function

The inference of unmarked function and register variation was investigated using traditional methods of calculation from categorical analysis techniques. Precision is the percentage of classifications made that were correct. Recall is the percentage of actual target classes that were correctly identified. An $F_{\beta=1}$ value is the harmonic average of the two.

The baseline here was defined as that given by an assumption of unmarked function, that is, the optimal result given by word order and part-of-speech.

It might be assumed that functions within a noun phrase are typically umarked. This work is the first empirical investigation of this assumption and shows it to be false: less than 40% of non-final adjectives realized Epithets; less than 50% of Classifiers were nouns; and 44% of Classifiers were marked. While the relative frequency of the various functions varied between registers (Munro, 2003b), the ratio of marked to unmarked function was consistent. The only functions with a $F_{\beta=1}$ baseline above 0.7 across all registers were Diectics and Things. For these two functions word order and close-group word-lists could have produced the same results without part-of-speech knowledge.

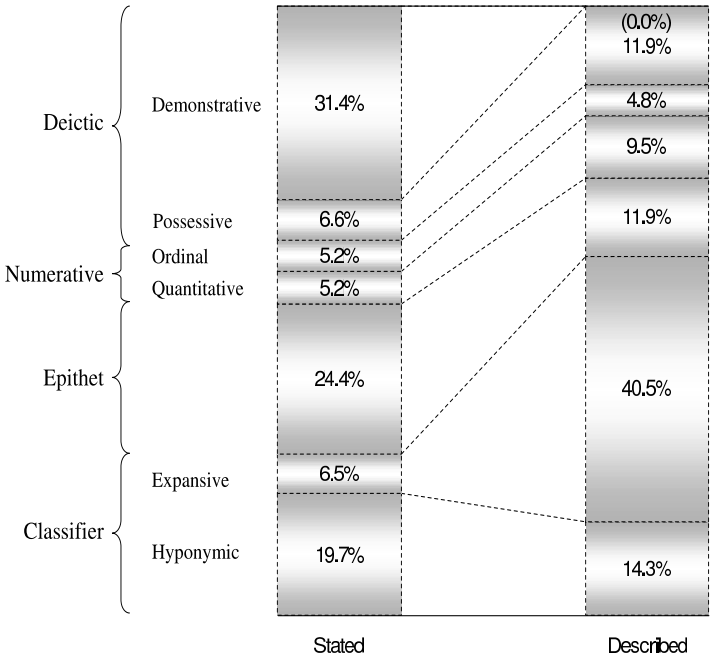

Figure 2: Delicacy and Choice in the modification of Stated/Described Things

All the registers had a $F_{\beta=1}$ baseline for Classifiers below 0.6, with all but the BIO-INF register below 0.5. For Epithets, Table 5 shows significant increases for all but Reuters-A, which has a surprising 0.101 loss in value. This was not a failure of *Seneschal* or the features but an artefact of the high precision for the baseline. An analysis of the data revealed cases such as 'Third time *lucky*', which was correctly identified as an Epithet in the unmarked case but misclassified by the learner. This was probably a generous allocation to the unmarked baseline as a post-modifying Epithet in English is very marked. The errors in the Numeratives primarily resulted from marked Classifiers ('the *100* metres') and Deictics ('*one* good reason'). The increase in the identification of Things was from the identification of multi-word Things, with collocation and repetition being significant features in the their disambiguation.

6 Conclusions

This paper has presented part of the results of the first attempt to learn aspects of a fundamentally probabilistic grammar within the framework of a Systemic Functional Grammar. It has been demonstrated that a machine learning representation of gradational functional with relatively low error is possible, and that such a representation is very rich in terms of the information it can provide for analysis.

The most significant outcome here is the level of sophistication in the discovery of the more delicate functions. As these were found through unsupervised methods there is the possibility of applying these methods to studies with much larger scope over larger scales, and building models towards describing lexis as most delicate grammar (Halliday, 1978; Hasan, 1987).

It is already the case that SFG parsers are more frequently used for linguistic analysis than for 'black-box' natural language processing. Hopefully, the work described here demonstrates that machine learning can also be a powerful tool and participant in linguistic analysis.

Function	Baseline			Classification			Gain / Loss
	recall	precision	$F_{\beta=1}$	recall	precision	$F_{\beta=1}$	$F_{\beta=1}$
All Registers:							
Deictic	97.7%	96.2%	0.969	97.5%	95.5%	0.965	−0.005
Quantifier	90.8%	56.4%	0.696	73%	78%	0.755	+0.059
Ordinative	80.3%	71%	0.754	77%	71.2%	0.740	−0.014
Epithet	79.9%	55.4%	0.655	67.2%	66.5%	0.668	+0.014
Classifier	60.1%	48.7%	0.538	65%	66%	0.655	+0.117
Thing	81%	98.9%	0.891	96.5%	96.8%	0.967	+0.076
Reuters-A:							
Deictic	100%	96.8%	0.984	99.2%	99.2%	0.992	+0.008
Quantifier	89.8%	58.7%	0.710	71.4%	85.4%	0.778	+0.068
Ordinative	69.6%	59.3%	0.640	79.2%	65.5%	0.717	+0.077
Epithet	78%	69.6%	0.736	63.4%	63.4%	0.634	−0.101
Classifier	66.1%	39.8%	0.497	71.4%	64.5%	0.678	+0.181
Thing	78.5%	99.3%	0.877	96.2%	97%	0.966	+0.089
Reuters-B:							
Deictic	97.4%	97.4%	0.974	97.4%	98%	0.977	+0.003
Quantifier	90.7%	68.4%	0.780	67.4%	90.6%	0.773	−0.007
Ordinative	87.1%	81.8%	0.844	74.2%	82.1%	0.780	−0.064
Epithet	65%	50%	0.565	57.5%	63.9%	0.605	+0.04
Classifier	75%	35.3%	0.480	70.8%	47.2%	0.567	+0.087
Thing	76.8%	97.4%	0.859	94.1%	95.3%	0.947	+0.088
Bio-Inf:							
Deictic	96.9%	97.9%	0.974	98%	83.5%	0.901	−0.073
Quantifier	92.5%	47.4%	0.627	85%	69.4%	0.764	+0.137
Ordinative	100%	0%	0.000	100%	0%	0.000	0
Epithet	82.6%	48.3%	0.610	72.5%	59.5%	0.654	+0.044
Classifier	54.5%	56.9%	0.557	61.5%	87.1%	0.721	+0.164
Thing	81.7%	99.5%	0.897	97.1%	97.1%	0.971	+0.074
Mod-Fic:							
Deictic	96.6%	93.5%	0.950	96%	99.3%	0.976	+0.026
Quantifier	88.9%	47.1%	0.615	55.6%	50%	0.526	−0.089
Ordinative	100%	62.5%	0.769	100%	71.4%	0.833	+0.064
Epithet	88.9%	61.5%	0.727	70.4%	84.4%	0.768	+0.04
Classifier	56.3%	36.8%	0.445	56.3%	37.5%	0.450	+0.005
Thing	87.3%	99.4%	0.929	98.9%	98%	0.985	+0.055

Table 5: Comparisons with Unmarked Function

7 Acknowledgements

Thanks to Geoff Williams and Sanjay Chawla. The experiments and most of the analysis was completed under their supervision.

References

B. Aarts. 2004. Modelling linguistic gradience. *Studies in Language*, 28(1):1–49.

R. Bod. 1993. Using an annotated corpus as a stochastic grammar. In *Proceedings of the Sixth Conference of the European Chapter of the ACL*.

B. Bohnet, S. Klatt, and L. Wanner. 2002. A bootstrapping approach to automatic annotation of functional information to adjectives with an application to German. In *The Third International Conference On Language Resources And Evaluation*.

J. Bresnan. 2001. *Lexical-Functional Syntax*. Blackwell, Oxford.

E. Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the first conference on North American chapter of the Association for Computational Linguistics*, pages 132–139. Morgan Kaufmann Publishers Inc.

M. Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania.

S.C. Dik. 1992. *Functional Grammar in Prolog: an integrated implementation for English, French, and Dutch*. Mouton de Gruyter, Berlin/New York.

R. Fawcett. 2000. *A Theory of Syntax for Systemic Functional Linguistics*. Philadelphia.

M.A.K. Halliday. 1978. *Language as a Social Semiotic*. University Park Press.

M.A.K. Halliday. 1994. *An Introduction to Functional Grammar*. Edward Arnold, London, 2nd edition.

M.A.K. Halliday. 2002. *On Grammar*. Continuum, London.

R. Hasan. 1987. The grammarian's dream: lexis as most delicate grammar. In M.A.K. Halliday and R. Fawcett, editors, *New Developments in Systemic Linguistics*. Pinter, London.

M. Johnson. 2003. Learning and parsing stochastic unification-based grammars. In *16th Annual Conference on Computational Learning Theory*, pages 671–683.

R. Kasper. 1988. An experimental parser for Systemic Grammars. In *Proceedings of the 12th International Conference on Computational Linguistics*.

P. Lane and J. Henderson. 2001. Incremental syntactic parsing of natural language corpora with simple synchrony networks. *Knowledge and Data Engineering*, 13(2):219–231.

W. Mann and C. Mattheissen. 1985. A demonstration of the Nigel text generation computer program. In J.D. Benson and W.S. Greaves, editors, *Systemic Perspectives on Discourse*, pages 84–95. Ablex, Norwood.

C. Manning. 2003. Probabilistic syntax. In R. Bod, J. Hay, and S. Jannedy, editors, *Probabilistic Linguistics*. The MIT Press.

C. Matthiessen and J. A. Bateman. 1991. *Text Generation and Systemic Functional Linguistics*. Pinter, London.

C. Matthiessen. 1995. *Lexicogrammatical cartography: English systems*. ILSP, Tokyo.

R. Munro. 2003a. Seneschal: classification and analysis in supervised mixture-modelling. In *Proceedings of the Third International Conference on Hybrid Intelligent Systems (HIS'03)*. IOS Press.

R. Munro. 2003b. Towards the computational inference and application of a functional grammar. honours thesis, Universtiy of Sydney.

M. O'Donnell. 1994. *Sentence analysis and generation: a systemic perspective*. Ph.D. thesis, University of Sydney, Department of Linguistics, Sydney, Australia.

M. O'Donnell. 1998. Integrating referring and informing in NP planning. In *Proceedings of the Coling-ACL '98 Workshop on the Computational Treatment of Nominals*.

T.F. O'Donoghue. 1991. A semantic interpreter for systemic grammars. In *Proceedings of the ACL workshop on Reversible Grammars*, Berkeley.

A. Ratnaparkhi. 1996. A maximum entropy part-of-speech tagger. In *Proceedings of the Empirical Methods in Natural Language Processing*.

G.H. Tucker. 1998. *The Lexicogrammar of Adjectives: a systemic functional approach to lexis*.