A Multi-Classifier Approach to Dialogue Act Classification Using Function Words

James O'Shea, Zuhair Bandar and Keeley Crockett

Department of Computing and Mathematics
Manchester Metropolitan University
United Kingdom
{z.bandar, k.crockett, j.d.oshea }@mmu.ac.uk

Abstract. This paper extends a novel technique for the classification of sentences as Dialogue Acts, based on structural information contained in function words. Initial experiments on classifying questions in the presence of a mix of straightforward and "difficult" non-questions yielded promising results, with classification accuracy approaching 90%. However, this initial dataset does not fully represent the various permutations of natural language in which sentences may occur. Also, a higher Classification Accuracy is desirable for real-world applications. Following an analysis of categorisation of sentences, we present a series of experiments that show improved performance over the initial experiment and promising performance for categorising more complex combinations in the future.

Keywords: - Dialogue Act, Speech Act, Classification, Semantic Similarity, Decision Tree.

Introduction

Collective Computational Intelligence is the form of intelligence that emerges from the collaboration and competition of many individuals. Whilst there is a natural tendency to focus on the machine aspects of multi-agent systems, ultimately these agents represent the beliefs desires and intentions (intentionality) of their human clients. It is difficult for the average human to express intentionality in the formal logic used by computers. Therefore Dialogue Management agents are required to bridge this gap, modeling user intentionality, conducting negotiations on the user's behalf and resolving conflict in the multi-agent system.

Dialogue Management (DM) is concerned with the communication between humans and computer-based systems using natural language. DM techniques support the production of Dialogue Systems (DSs) which will allow ordinary users to interact with increasingly powerful and complex applications in the future. Dialogue Act (DA) classification is an established element of research in the Natural Language Processing (NLP) approach to DM [1-6]. DA theory asserts that a sentence or spoken utterance can be separate into two components, the Propositional Content (i.e. what it is about) and the DA (i.e. what is it is saying about the propositional content) [7]. So the propositional content "door is shut" can be in a question DA "Is the door shut?", an instruction DA "Shut the door!" or an assertion DA "The door is shut.", which mean quite different things.

This paper extends an investigation of the hypothesis that DAs can be classified into different categories solely by using function words. An initial study using the technique achieved a Classification Accuracy (CA) of 89.43% when classifying questions against a challenging mixture of non-questions [8]. Two research questions emerged from this work, "How can the CA be improved to the point where it is useful in real-world applications?" and "Can even more challenging combinations of questions and non-questions be classified effectively?"

More challenging combinations arise because questions and non-questions each come in a variety of forms. A generic classifier may be confounded by the fact that features suitable for discriminating between a particular form of question and a particular form of non-question could become obscured in the general mix. This paper extends the technique by investigating the production of specialist classifiers for particular combinations of forms of question vs. non-question DAs. A user utterance could be processed by a bank of specialised classifiers running in parallel which would collectively decide whether a question was present, and if so, its position in the utterance. This collective approach is also expected to be capable of dealing with complex utterances containing multiple, different CAs as classifiers are also under development to discriminate between instructions and non-instructions [9].

One example of the value of classifying DAs is in the use of questions, instructions and assertions to communicate with robots [10]. Another important application for DAs is in providing humans with advice in complex areas such as debt management and on workplace bullying and harassment procedures [11]. A new potential class of applications is also emerging, which may be described as Mandated Intentional Agents (MIAs). The concept of an Intentional Agent, acting according to its own beliefs, desires and intentions in the world, is an established concept in cognitive science [12]. An MIA is a machine-based agent which represents the beliefs, desires and intentions of a human client in the real world. One potential application would be buying and selling electrical power on behalf of a household connected to the smart grid [13]. Such an agent would be able to make advantageous deals on behalf of its human owner at any time of day or night as opportunities arose. The best method to instruct an MIA in such a complex task would be natural language dialogue.

The majority of current DSs use Pattern Matching (PM) or NLP to analyse and answer a user utterance. PM systems are considered to be the best for developing DSs that seem to be coherent and intelligent to users [14]. They support scalability to large numbers of users because they do not require pre-processing stages, but they are labour intensive to develop and maintain. NLP systems have a substantial theoretical basis but require a chain of computationally intensive and error-prone stages such as pos-tagging, syntactical repair and parsing. This rules them out for web-based systems that must service many users in real time.

Short Text Semantic Similarity (STSS) offers an alternative approach to PM and NLP. A user utterance (a unit of dialogue containing a communicative action [1]) is acquired as a Short Text (ST) and compared with a set of prototype STs. The ST with the highest similarity to the user utterance is taken to be its meaning and triggers a suitable response from the DS. However, current STSS algorithms have a weakness; they are oblivious to the DA performed by an utterance.

Latent Semantic Analysis (LSA) is a well-known method of measuring semantic similarity [15]. Taking the examples listed earlier, LSA scores all of the possible pairwise combinations of question, instruction and assertion as 1.0 – i.e. identical in meaning. Therefore devising an efficient method of classifying Dialogue Acts (DAs) will be a crucial first step in measuring the true semantic similarity between a pair of STs accurately.

The rest of this paper is organised as follows: section 1 briefly reviews prior work on function words, DA classifiers and evaluation methods. Section 2 describes the classes of the questions and non-questions used, how they were collected and how the training and testing files were composed. Section 3 outlines the experimental procedure, section 4 discusses the results and section 5 contains conclusions and recommendations for future work.

1 Prior Work

1.1 Features for DA Classification

Machine classification of DAs has used various techniques including n-gram statistical models [2], Bayesian networks [16] and Decision Trees (DTs) [3]. All of the approaches rely on extraction of features from the DA to present to the classifier.

The most common feature used is the n-gram. An n-gram is a sequence of contiguous symbols found inside a longer sequence. In DA classification it is usually a short string of words extracted from a marked up corpus [2, 4]. The importance of the n-gram is that any n-gram could be a predictive feature for classifying the text containing it.

Large numbers of n-grams are generated by any real-world corpus so a subset is used (in [3] sets of 10 or 300). N-grams have been used to code features for Hidden Markov Models [4]. Bigrams such as CanYou and IWant [16] have been used in Bayesian classifiers, which also combined them with the DAs of previous utterances in the dialogue [1].

Cue phrases have been used as features for a simple, efficient classifier described by Webb et al [2]. A cue phrase is a longer form of n-gram, selected on the basis of predictability for particular DA classes. In operation, the classifier examines an utterance for cue phrases and assigns the class of the highest predictive phrase it contains. This technique has the serious disadvantage that it ignores other potentially valuable information in the utterance.

In the LSA approach to DA classification [5], the features are terms in a “query” vector. A cosine measure is used to determine the vector of the closest matching “document” and the DA type of the match is assigned to the query. LSA normally removes function words [15], but it is not specified if this is the case with DA classification.

Keizer et al [16] used 13 surface features to train DTs in a comparison with a Bayesian network. Features included length, various starting bigrams, presence of particular words, positive/negative references and presence of particular grammatical classes. Keizer's work is interesting because it classifies DAs into two classes, forward-looking functions (acts that have an effect on following dialogue) or backward-looking functions (that relate to previous dialogue).

The closest work to this paper [3] uses a decision tree trained with a mix of features including the presence of a question mark, a count of the occurrences of the word OR, utterance length, a bigram of the labels of the last two DAs, n-grams of pos-tags for the utterance, n-grams of words in the utterance and the top 10 predictive individual words. A review by Verbree et al [3] also refers to the use of verb type and cites a number of uses of prosodic features.

These approaches have achieved good results, but they use complex and computationally intensive feature extraction which rules them out from future real-time applications. Also, Cue Phrases may discard considerable useful information.

1.2 Function Words

Words in the English Language can be divided into Function Words (a closed class of structural words such as articles, prepositions, determiners etc.) and Content Words (the open classes of nouns, verbs, adjectives and adverbs). "Stop word" lists contain words with a high frequency of occurrence - mostly function words with a few high frequency content words mixed in. There is no definitive list of stop words, although one by van Rijsbergen [17], which is often cited, contains 250 words. Others posted on the web contain over 300 [18]. A set of 264 function words (available from the authors on application) was compiled for this paper by combining stop word lists, removing the content words and then adding low-frequency function words from dictionaries.

Information Retrieval (IR) places a greater value on content words than function words in searching for documents. Spärck-Jones' [19] TD/IDF approach and Salton's [20] Vector Space Model increase the contribution of low frequency words to the similarity measurement. LSA removes 439 stop words [21], having "little or no difference on the SVD solution." [15] from the submitted terms or documents before comparing them. This list has not been published. Citeseer also removes stop words prior to performing word frequency calculation [22].

The STASIS [23] STSS measure used function words as well as content words because they carry structural information which is useful in interpreting sentence meaning [23]. However, STASIS can only detect matches between identical pairs of function words in the two STs. Recently an STSS algorithm which makes use of a corpus-based measure combined with string similarity, but filters out function words [24], has been reported to achieve good performance.

1.3 The Slim Function Word Classifier for Dialogue Acts

We classify DAs at a coarse level of granularity as we believe distinguishing between questions, instructions and assertions will be most useful in practical DSs [11] and Robotics applications [10]. This work uses the Slim Function Word Classifier (SFWC) approach which takes a radically different view of the value of function words. The SFWC assumes that sufficient information is contained in the function words of an utterance alone to allow its DA to be classified [8]. Consequently, each function word in the utterance is replaced with a unique token and all content words replaced by the same wildcard. Table 1 shows an example of a tokenised question.

Table 1. Tokenisation of a typical question

Question	**does wearing caps or hats contribute to hair loss**
Tokenised form	**56,0,0,156,0,0,212,0,0,300,300,300,300,300,300,** **300, 300, 300, 300,300, 300,300,300,300,300**

Processing a short text requires two stages. The first is the expansion of contractions. This uses a lookup table to replace contracted forms such as Don't with their full forms, i.e. Do not. Some

contractions are ambiguous e.g. She'd could be She would or She had. At this stage of the work we take a brute force approach of replacing all the variants with a single form (e.g. 'd forms with would). Apostrophe usage in forming possessives is ignored and all possessives are treated as content words.

Second, the words comprising the sentence are successively looked up in a table of function words and numeric tokens (composed from the list described in 1.2). If the word is found it is replaced by the appropriate token (range 1 – 264). In this experiment the tokens are allocated in ascending order to the alphabetically sorted list of words, so that 1 represents the word "a" and 264 represents "yourselves".

If the word is not found it is replaced by the token 0 (indicating a content word). If the sentence contains fewer than 25 words, all empty slots are filled with a "no word present" token given the value 300. This particular value was chosen so that it would be easy for the DT algorithm to partition it from the word tokens.

Note that all punctuation, apart from the possessive apostrophe is stripped out as part of pre-processing. This means that the presence of a question mark – which has been used as a feature in some studies [3] is not used.

The brute force approach taken by the SFWC algorithm is particularly efficient in using only simple table lookup. It avoids complex pre-processing as parsing-based disambiguation would increase the computational load greatly. This virtue is emphasized because the same pre-processing steps will be required for data entered into real-world systems deployed on the web, which must be scalable to large numbers of users in real-time.

It should also be observed that the tokenisation generalises the sentences when they are preprocessed, i.e. two different questions could be based on the same skeleton of function words, so that after tokenisation (when all of the content words are replaced by zeros) they are represented by the same token string.

1.4 Evaluation Datasets

The normal method of evaluating classifiers is to measure their CA on test data which was not used to train the classifier. Comparing DA classifiers is difficult as there is no standard dataset for developing and testing them. The different datasets used are tagged with different numbers and types of DA [3], ranging from 2 [16] to 232 different tags [5]. The sizes of the datasets also vary, from 223,000 utterances in the Switchboard set [2] to one of 81 utterances for a DT classifier [16]. Variety is also an issue. One dataset of 2794 DAs was drawn from only two humans [4] and may not generalise beyond these two people. One example of very good performance was a CA of 89.27% for a classifier with over 40 DA categories using a large training set [3]. Results can be variable however; another study [6] achieved 92.9% on suggestions but scored 0 on queries.

The first stage of the present work used the particular task of discriminating between Questions vs. Non-questions. This was because prior empirical experience with DSs has shown this to be a non-trivial problem and because the task is appropriate to other agent-based applications such as robotics.

Data is required for training and testing Artificial Intelligence (AI) classifiers. In this case it was decided to collect a new dataset, principally because existing datasets are not appropriate for the instant messaging style dialogue expected in DS applications. Many datasets examined from the literature are terse, having been derived from telephone-based Automatic Speech Recognition systems and may be constructed to test aspects such as repair of misunderstood utterances. It was also decided to collect 3 types of utterance: straightforward questions, straightforward non-questions and difficult non-questions. It will be shown in section 2 that a representative range of question and non-question sub-categories can be constructed from this base set.

Choice of the size of the dataset was influenced by three interacting factors: the effort involved in collecting the data, the minimum size required to represent the domain and the execution time required to construct a decision tree. These factors were balanced by using n-fold cross validation. Under n-fold cross validation, the dataset is randomly divided into n folds. Working through the folds in sequence, one is held back for testing whilst the others are used for training. This allows all of the data to be used for training and testing a large set of classifiers, without any of the classifiers being tested on the data it was trained on. All of the following trees were trained using 60-fold cross validation on a set of 600 questions and 600 non-questions. Choice of 60-fold cross validation meant that over 1,000 training cases were used to construct each classifier (a rule of thumb recommended by Quinlan for Decision Tree classifiers [25]). Also, this allowed reasonable execution times for the training and testing of the classifiers. Finally, manually mining the Web for a pool of questions and non-questions to compose the datasets was feasible at this size. Some classifiers require fixed length records, so an upper limit of 25 words was set (determined empirically).

The first step was the collection of a pool of "straightforward" questions and non-questions. Questions were acquired from FAQ lists and non-questions were acquired from blogs. Both of these sources are capable of producing texts which are like the utterances used on messaging-style dialogue. 1,660 straightforward questions were collected from the highest user-rated FAQ lists from the Usenet news system and 2288 straightforward non-questions were collected from "blogs of note" commended blogs on blogspot.com.

Straightforward questions are those which are fairly easy to recognise as they often have distinguishing features such as a wh-cheft (what, who) or an auxiliary verb (can, is) at the beginning. The straightforward non-questions are sentences which are not questions and do not have one of the question's distinguishing features at the beginning. To increase the challenge of the dataset, difficult non-questions were included. A difficult non-question is defined as a non-question which starts with one of the question's distinguishing features. For example the word "What" is commonly used to start a question and in this role it is known as an interrogative introducer. For example:

What alternative therapies exist and are they any good?

On the other hand, "What" can also be pronoun, in which case it is not indicative of a question. For example:

What all men want is beer and sport.

In fact non-questions rarely start in this way and it was not practical to collect them from web sources, so a set of additional non-questions was synthesized. Despite their rarity, we have observed empirically that confusion of this type of utterance with questions reduces users' confidence in the agent. The approach of synthesising rare data is well-established [26]. Sometimes the synthetic non-questions are not valid sentences in their own right, but they would make sense as a terminating utterance in a dialogue sequence.

The dataset for the initial experiment [8] was composed in the approximate proportions 50% questions, 25% straightforward non-questions and 25% difficult non-questions (using random sampling). In fact the final proportions in the dataset were 591 questions and 615 non-questions. For each category, a randomly selected sample larger than the target size was taken. The sentences were then pre-processed and duplicates removed to create the initial dataset. Because there were more duplicates in the questions and fewer in the non-questions the classes were slightly imbalanced. Examples from the dataset are shown in table 2.

Table 2. Example training data

Category	**Example**
FAQ question	**Does wearing caps or hats contribute to hair loss?**
Blog non-question	**Sometimes the psoriasis treatment causes the hair loss.**
Synthetic non-question	**Which in many cases can be cured with a simple lotion.**

The pool of 1,660 straightforward questions and 2288 straightforward non-questions was also used as the source material for the new experiments reported in this study. However, in this case, after tokenisation the datasets were balanced so that exactly 600 of each were obtained. This involved an iterative process of adding or removing cases from the dataset followed by re-tokenising until all the duplicates were replaced.

It should also be noted that questions were balanced by non-questions derived from the same context. Contexts were highly varied including "IRS", "Immunisation", "Tattoo" and "Gasoline." The majority of non-questions were assertions (statements, clarifications and answers to previous questions) with some instructions.

1.5 Choice of classifier

A good cross-section of classifiers has been evaluated for DA classification. These include statistical [27] and n-gram models [2], Bayesian networks [16], Naïve Bayesian Classifiers [28], Kohonen Networks [29], Multi-Layer Perceptron [28], Backpropagation Artificial Neural Networks [30], Maximum Entropy [31], C4.5 Decision Trees [3], Production Rules [32], Simple Heuristics [33], Hidden Markov Models [34], Partially-Observable Markov [35], K-Nearest Neighbour [30], Support Vector Machines [36], Learning Vector Quantisation [37] and Self Organising Maps [37].

Unfortunately, it is difficult to draw any conclusions about the relative performances of particular classifiers, because different studies use different features, different DA taxonomies and different corpora – so results are not comparable. The initial investigation conducted for this study [8] used 4 of the most common techniques, Decision Trees (C4.5), Naïve Bayes, Bayesian Classifiers and Multi-Layer Perceptrons (MLP).

DT induction is a highly effective method of machine learning for classification. One of the most well-established algorithms is C4.5 [25]. DTs partition the sample space recursively and the outcome is a set of rules induced from a training set of example instances previously labelled with their classes. The chief advantage of DTs over other classifiers is that the rules "explain" how they reach their decisions and (combined with pruning) provide a greater insight into the problem domain.

The starting point of a DT is one node with all examples labelled as belonging to one class. A node is 'impure' if the examples reaching that node are not all in the same class. During training impure nodes are successively partitioned to create new, purer nodes. The final, leaf, nodes are labelled with the appropriate class. Impure leaves, which may occur if the attributes have been tested exhaustively without reaching purity, are labelled with the majority class. An alternative pruning technique, Minimum Number of Objects (MNO) removes leaves with fewer than a specified number of cases and re-distributes them amongst leaves at the level above in the tree.

Bayesian techniques, such as Naïve Bayes and the Bayesian Network have long been of interest to NLP researchers due to their statistical origins. The Naïve Bayes classifier uses information extracted from a set of example instances to produce a probability estimate of the class of a new instance, assuming statistical independence between the attributes to make the estimate [38]. The Bayesian Network is an alternative to decision trees, which uses a Directed Acyclic Graph structure [38]. This structure is less constrained in allowing linkage between nodes than a pure tree. Also, the node does not contain a splitting rule; rather it contains a probability distribution that is used to predict the class probability for a particular instance. The proposed advantage of the Bayesian Network over the DT is that it retains information that is lost due to splitting in DTs. However, the Bayesian network is also sensitive to missing values and this is an issue in handling STs of varying lengths.

The Multi-Layer Perceptron (MLP) is an Artificial Neural Network (ANN), modeled on the biological structure of the brain [39]. Inputs are fed through a network of simple processing units with weighted connections to produce (in this application) a classification output. The MLP is trained using backpropagation, which uses a training set to modify the weights until the performance is optimised. Of the 4 techniques, the MLP is the least transparent in explaining its decisions; however it has the benefit of robustness to noise and missing values.

The initial evaluation of the classifiers was performed using the WEKA data mining tool. All comments about statistical significance are based on the standard test used by WEKA, the corrected re-sampled t-test. Table 3 shows the best Classification Accuracies obtained, using these classifiers at their default settings in WEKA. Results are also shown for the ZeroR and OneR classifiers, which provide a benchmark for the performance of the other classifiers. ZeroR shows the CA if all of the cases were labelled with the classification of the majority class, and helps the interpretation of unbalanced data. In this initial set of experiments, the dataset was slightly unbalanced, so that labeling all data with the majority class gives a CA of 50.99%. The OneR classifier shows the CA obtained using the single best classification rule that could be found. This is intended to be an indicator of the complexity of the domain. In this case a CA of 82.06 indicates that it is quite easy to classify four fifths of the cases, however it does not tell us how difficult it will be to improve on the classification of the remainder.

Table 3. Classification accuracies for baseline measures and popular classifiers

Classifier	**ZeroR**	**OneR**	**Naïve Bayes**	**MLP**	**Bayes Net**	**C4.5**
CA	**50.99**	**82.06**	**55.98**	**69.14**	**77.87**	**88.73**

Each increase in CA across the AI classifiers is statistically significant. All of the classifiers outperformed the ZeroR baseline significantly. C4.5 also outperformed the OneR classifier significantly, whereas each of the other techniques failed to reach the OneR level of performance.

A further set of experiments was conducted to investigate optimisation of the C4.5 decision tree, using two different methods of pruning, Confidence Interval (Conf) and Minimum Number of Objects (MNO), starting from the baseline of 88.73% CA and a DT size of 118 nodes

Optimising CA, confidence pruning achieved a best CA of 89.43% (Conf = 0.04) and MNO pruning achieved a best CA of 88.97% (MNO = 5). On the other hand, when optimising (minimising) tree size, confidence pruning achieved a size of 47 nodes (Conf = 0.0003) and MNO pruning achieved a size of 46 nodes (MNO = 13). These were the greatest reductions that could be achieved before a statistically significant reduction in CA from the baseline value of 88.73%. This degree of pruning is a good indicator that the DT classifier is generalising to model the domain effectively, rather than just memorising the training cases.

The high performance of DTs, combined with their efficiency and transparency, was conclusive in their choice for further experiments.

2 Classification of questions and non-questions

2.1 New question sub-classes

An investigation of question taxonomies revealed two useful approaches to categorising questions: Grammatical classification [40] and Domain-based classification [41] [42]. Additionally, the highest level of grammatical distinction between different types of sentences is between Simple and Multiple sentences.

2.1.1 Grammatical Classes of questions composed of Simple Sentences

A Simple Sentence contains a single independent clause. The standard grammatical classes of questions derived from simple sentences, with an example in each case [40] are listed below:

1. Yes-No
   Have you finished the book?
2. Wh-questions
   What is your name?
3. Option selection
   Would you like to go for a walk or stay at home?
4. Tag questions
   They forgot to attend the lecture, am I right?
5. Declarative questions*
   You've got the explosive?
6. Exclamatory*
   Hasn't she grown?
7. Rhetorical*
   Who cares?

* Without prosodic information the declarative question form would be interpreted as a declarative rather than a question. Declarative, exclamatory and rhetorical are all dependent on prosodic information for their correct interpretation. This can be resolved using the question mark. As Quirk

observed [40] "… the question mark matches in writing the prosodic contrast between this sentence as a question and the same sentence as a statement." The exclamatory and rhetorical categories do not implicitly ask for information; uses include emphasis or giving an opinion. Given the text-based approach, declarative, exclamatory and rhetorical questions are not suitable for inclusion in this study.

2.1.2 Grammatical Classes of questions composed of Multiple Sentences

Grammar recognises two more sophisticated question categories which could be more challenging to the techniques developed in this work – these are questions embedded in Compound Sentences and Complex Sentences.

A Compound sentence contains two or more co-ordinated main clauses, for example:

I admire her reasoning but were her conclusions valid?

A complex sentence has a single main clause and one or more subordinate clauses, for example:

Can you confirm which flight we are taking?

The complex form allows questions to participate in indirect DAs, for example:

Please confirm which flight we are taking.

is a directive (instruction) which seeks the same information as the question Which flight are we taking?

2.1.3 Domain-based classification

An example of domain-specific categorisation is the assessment of medical students. A list of categories, with examples, derived from Christensen [42] is shown below:

1. Information-Seeking Questions:
What were the blood values from the lab?
2. Diagnostic Questions:
What conclusions did you draw from these data?
3. Challenge (Testing) Questions:
What evidence supports your conclusion?
4. Hypothetical Questions:
If the liver function tests were normal, how would that have affected your treatment plan?
5. Action Questions:
What needs to be done to implement the plan for this patient?
6. Extension Questions:
What are the implications of your conclusions for the treatment of asthma among children in elementary school in our community?
7. Priority/Sequence Questions:
Given the patient's limited resources, what is the first step to be taken?
8. Prediction Questions:
If your plan (conclusion) is appropriate, what do you expect to happen over the next month? Year?
9. Generalisation Questions:
Based on your experience and the studies of the incidence of teenage pregnancy, what do you consider to be the most effective strategies for our local high school teachers and counsellors?

Of the above categories, 1,2,3,5 and 6 are questions comprised of simple sentences and 4, 7, 8 and 9 are complex questions. From the point of view of constructing a dataset for this study, there is nothing to be gained from discriminating between questions such as What were the blood values from the lab? and What evidence supports your conclusion?

Some other types of question were found during the investigation:

- indirect questions:
  Joan asked was he ready yet?
- directives containing questions:

Please confirm which flight we are taking.

- assertives implying questions:
  I have received an invoice and I do not know what it relates to.
- leading questions:
  You did have the gun when you left, didn't you?

The leading question is a specialist form from the legal domain and is easily disposed of. Its distinguishing feature is that it implies a correct (or known) answer. Structurally these questions will fall into one of the categories described previously and therefore will be covered automatically (the example given is a negative form of tag question). The indirect questions don't necessarily increase the complexity of the recognition task over and above the existing categories. The problem is deciding what to do in response – how they would affect a Conversational Agent's intentionality within its task and problem domain. This is beyond the scope of the present work so this question form will not be included.

The Directive corresponds to Which flight are we taking? and the Assertive corresponds to Can you explain what the invoice I have received relates to?

In fact, both are indirect dialogue acts; they seek information but are prima facie, non-question. As such they will not be included in the current work (although the examples look promising in terms of the function words contained in the sentences).

2.2 Classes of questions and non-questions used in datasets

The classes described in section 2.1 indicate what should be represented in the combinations of questions and non-questions used in this chapter. However, a one-to-one mapping between the classes in 2.1 and the datasets used for training and testing is not required as some of the grammatical or domain-based categories are indistinguishable for the purposes of this study. The categories described in the following sections were derived for experimental purposes.

2.2.1 Straightforward questions

The Straightforward question type is, effectively the question contained in a Simple Sentence. The term "Straightforward" has been adopted to indicate that although the question itself may be quite sophisticated or difficult to answer, the form of the question is likely to be the least challenging for a DA classifier.

These questions are short and to the point. They do not require resolution of references to prior dialogue and they do not contain pertinent information embedded in clauses separated from the main question clause. Good examples of this simplest form, such as "When was James Dean born?" can be found in the TREC factoid set. Such questions have a very obvious feature in the first word position – the presence of a Wh-cheft.

Straightforward questions cover the grammatical classes Yes-No, Wh-questions and Option selection from the simple sentences class and provide evidence for some questions from the multiple sentences class. For example, they will provide evidence that the first question in a compound question will be recognised. They also cover the domain-based classes Information-Seeking, Diagnostic, Challenge, Action and Extension.

2.2.2 Straightforward questions with preambles

It is possible to create a more difficult class of questions by shifting the first word (typically an auxiliary verb or a wh-cheft) of the question further down the sentence. This can occur in a form known as the "Pushdown" where a phrase or part of another clause, is moved to precede the main clause e.g. On what side of the road was he driving?

Introductory words and phrases, which do not qualify as clauses in their own right, can also be used with a question. They have a variety of purposes which don't actually contribute to the semantics of the sentence, for example for continuity, politeness, attention grabbing etc.

These words and phrases are accommodated in this study by adding a "preamble" of a few words to the start of the sentence. For the purposes of this study a preamble is any phrase of up to a maximum of 5 words and specifically does not contribute to the semantic content of the utterance. Thus examples include:

Actually

Almost everyone asked
And there is another thing

So a question of this kind would be

And there is another thing, when was James Dean born?"

These preambles look remarkably similar to Cue Phrases, however the two properties of a good preamble are that (i) it must "work" when placed in front of a question (like a cue phrase) and (ii) it must not provide evidence to the classifier that it is part of a question (contrary to a cue phrase). The middle phrase, *Almost everyone asked*, may appear to contradict these requirements. However, the key evidence for a question is in the word *asked*, a verb which will not be considered by the SFWC. When producing preambles for non-questions it was fairly straightforward to replace preambles of this type with equivalents that worked for non-questions but generated the same tokens. For example, *Almost everyone denied*.

Straightforward questions with preambles cover short Tag questions from the grammatical class and short versions of the Hypothetical, Priority/Sequence Prediction Questions and Generalisation questions from the domain-based classes.

2.2.3 Simulated clauses

The need to represent the grammatical category of Questions in Multiple Sentences posed some problems. One source of data, the IRS FAQ set [43] contained some questions fitting this form:

For business travel, are there limits on the amounts deductible for meals?

If I claim my daughter as a dependent because she is a full-time college student, can she claim herself as a dependent when she files her return?

I received a Form 1099-MISC instead of a Form W-2. I'm not self-employed, I do not have a business. How do I report this income?

The worst case of the last form of question is one that tends to occur in computer helpdesk applications, which contains a relatively long description of the current state of a computer followed by "…Can you help?"

Preliminary work revealed that attempting to collect examples of this data from real-world dialogue sources would be very heavily time consuming and have a low productivity. However, it was also clear that for this category, the questions were again straightforward questions with one or more clauses prefixing them. Therefore, for this set of experiments it was decided to prefix questions with non-questions to simulate the clauses. This may provide a harder classification task than real life because clauses that naturally precede a question may contain additional semantic features.

The IRS source is particularly interesting, because much of the previously reviewed work on DA classification concerned itself with problems involved with spoken dialogue such as dealing with incomplete utterances and channel management. However, when these are resolved the dialogue content is relatively simple. The IRS site reveals that even when these modal problems of dialogue have been resolved, the underlying problems of communicating in a real-world goal-oriented system are rich and complex.

Simulated clauses cover longer Tag and Multiple Sentences questions from the grammatical class, in particular the second question of a compound question.

They also cover longer versions of the Hypothetical, Priority/Sequence Prediction Questions and Generalisation questions from the domain-based classes.

2.2.4 Omitted question classes

The grammatical class Declarative was omitted because the only way it can be identified is with prosodic information. Prosodic information is beyond the scope of this study, which deals with text-based input. However, this information could be used in future DT classifiers.

Some questions composed of multiple sentences have also not been covered. In particular, a question in an Indirect DA could confuse a single DA classifier. However, it is anticipated that future multi-classifiers could cope with this, for example the sentence:

Please confirm which flight we are taking.

could be classified as Instruction containing a Question.

2.2.5 Straightforward vs. Difficult non-questions

Two basic forms of non-question are required for this study, straightforward and difficult. Further sub-categories may be generated by applying the variants such as preambles and simulated clauses devised for the questions.

The problem of "Difficult non-questions" arises from testing by Donald Michie of early pattern matching systems developed by the MMU Centre for Conversational Agents. Simple patterns based on the occurrence of Wh-chefts break down when non-questions are framed using the Wh-chefts as pronouns or conjunctions. In particular, it is possible to construct valid non-questions with pronoun Wh-chefts or conjunctions as starting words.

An example taken from a hair / beauty blog site (http://www.kaboodle.com/reviews/psoriasisnet-6) is:

When psoriasis develops on the scalp, hair loss sometimes follows.

and this needs to be represented in training and testing data used in the following experiments.

The term "difficult" is used because although sentences of this form do not have to be complex in a cognitive sense, the wh-cheft in the first position makes it more difficult for a DA classifier to deal with correctly. A straightforward non-question on the other hand does not contain features which suggest a question in the first few words.

As with the questions, varying forms of difficult non-questions can be generated from this starting point. Using this type of data calls for some fine judgment. The problem is that it is possible to construct some really taxing non-questions using wh-chefts as pronouns (and find other function word usages that cause problems). However, of their nature some of these are quite unnatural sounding and would be of very low frequency in a natural interaction. So the dilemma arises:

- Should these forms be included in a dataset at all when they are likely to occur with low frequency?
- If so should they be represented in proportion to their frequency of occurrence in the language?

The decisive factor for this work was "Loebner behaviour." This is most prominent in the logs of judging of the annual Loebner prize, although it has been observed to a lesser degree in logs from Conversational Agent tests conducted by the MMU Centre for Conversational Agents. When a human uses a Conversational Agent, the interaction tends to proceed without incident as long as the agent behaves like a human. As conversational partners (particularly in goal-oriented dialogue) both humans and Conversational Agents will make mistakes. As long as the agents make human-like mistakes things go reasonably well. However, even if a Conversational Agent is outperforming the equivalent human in terms of things like domain knowledge, once it makes the kind of mistake a human would not make the user tends to fasten on this to the exclusion of achieving the goal. Judges in the Loebner prize competition tend to be particularly vicious and relentless in this kind of behaviour. So exploring the issues of "difficult non-questions" is important if the product of the DA classifier is to be deployed ultimately in a Conversational Agent.

Having decided to include them, it is necessary to include them in far greater proportions than occur in real-life, otherwise the classifiers will not train properly. Therefore it should be born in mind that classifiers for exotic combinations that have a lower CA are not likely to degrade the general performance of overall DA classification significantly.

3. Experiments

Designing the experiments required the formulation of hypotheses followed by creation of suitable datasets, then the training and testing of decision tree classifiers. Again, these experiments were performed using WEKA and all of the comments about statistical significance are derived from the

corrected re-sampled t-test. In the following experiments Classification Accuracy is the percentage of correctly classified DAs (both Question and Non-question) from the whole test data set.

3.1 Hypotheses

Table 4 shows a number of alternative hypotheses to be tested, for different combinations of question and non-question sub-categories. In each case the null hypothesis is that the classifiers do not score higher than chance in discriminating between questions and non-questions in the various combinations. Each hypothesis requires a series of experiments to be conducted to test it, so that when cross-validation and repeated runs from randomized starting points have been taken into account, in excess of 62,400 decision trees were constructed for the new experiments described in this paper.

Table 4. Experimental hypotheses

Experimental series	Hypothesis (H_1)
1	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions vs. straightforward non-questions.
2	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions with 1 word preambles vs. straightforward non-questions.
3	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions with 2 word preambles vs. straightforward non-questions.
4	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions with 3 word preambles vs. straightforward non-questions.
5	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions with 1-3 word preambles vs. straightforward non-questions.
6	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions vs. straightforward non-questions when both have 1 word preambles. (The decision tree does not learn to classify more effectively by learning from features in 1-word preambles)
7	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions vs. straightforward non-questions when both have 2 word preambles. (The decision tree does not learn to classify more effectively by learning from features in 2-word preambles)
8	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions vs. straightforward non-questions when both have 3 word preambles. (The decision tree does not learn to classify more effectively by learning from features in 3-word preambles)

9	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions vs. difficult non-questions.
10	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions vs. difficult non-questions when both have 1 word preambles.
11	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions vs. difficult non-questions when both have 2 word preambles.
12	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions vs. difficult non-questions when both have 3 word preambles.
13	A decision tree using function words can achieve classification accuracy significantly higher than chance over the dataset of straightforward questions vs. straightforward non-questions when both are preceded by difficult simulated clauses.

Experiment 1, in discriminating between straightforward questions and straightforward non-questions, is expected to pose the simplest discrimination task and set the upper bound of expected CAs for the following experiments. It also represents the kind of challenge a DS should face if the human user is complying with Grice's rules, in particular that dialogue should be clear, direct and to the point. [44].

Experiments 2-5 take the straightforward questions from the initial dataset and shift the first word position progressively further down the question. The preambles may contain a mix of function and content words, potentially obscuring an important feature. Experiment 5 provides a baseline generic classifier against which the performance of specialized classifiers 2-4 can be measured.

Experiments 6-8 correspond to experiments 2-4. These experiments were performed to check whether there were common features in the preambles which contributed to the classification, confounding the results of experiments 2-4.

Experiments 9-12 investigate how effectively the classifiers can work when features in the question first word position are counterbalanced by similar features in the non-question first word position. The difficult non-questions all start with a word which normally signifies a question when it appears in the first word position. Experiment 9 corresponds to experiment 1 and provides a baseline for comparing the effect of adding preambles in 10-12. Experiments 10-12 also correspond to experiments 6-8 by inserting a preamble in front of the original first word position.

3.2 Experimental procedure

For each of the hypotheses listed above, a series of 4 experiments was conducted to determine the highest classification accuracy obtainable and the optimal level of pruning. The highest CA is an indicator of the performance of a trained classifier deployed in an application. The degree to which a decision tree classifier can be pruned before the CA drops significantly provides evidence about the degree to which the tree can generalise to model the problem domain.

Each experiment consists of a series of trials varying a DT pruning parameter. The first series of trials uses a standard initial set of confidence intervals to control the level of pruning, which establishes approximately where the optimum pruning level will be found. The second series of trials explores a range of pruning values about this initial approximation to establish optimal pruning. These two series usually establish both the maximum CA and the optimal pruning level; but if not, further series may be run based on the information obtained from them.

Confidence interval pruning is based on comparing the expected error rate for the original subtree with that for the replacement node. "Expected error rate" refers to the error rate that would be expected

if the tree were run with an independently selected test dataset. The actual value is not known during tree construction, but the confidence interval defines the range it would be expected to fall in [25].

The third series of trials uses a standard initial set of values for Minimum Number of Objects (MNO) pruning. MNO pruning sets a minimum number of training cases for each leaf node. Again the fourth series of trials pins down the optimal pruning level for MNO pruning. Detailed results for series 1 are given in tables 5-8. In the tables a significant decrease in CA is marked with an asterisk.

Table 5. Experiment 1.1

Conf	0.25	0.2	0.15	0.1	0.05
CA	98.50	98.51	98.41	98.48	98.48
Tree Size	29-71	29-67	29-47	29-33	29-33

Table 6. Experiment 1.2

Conf	0.0005	0.0004	0.0003	0.0002	0.0001
CA	98.36	98.36	97.88*	97.89	98.77
Tree Size	25-31	25-31	21-31	21-31	17-31

Table 7. Experiment 1.3

Min	2	5	10*	15	20
CA	98.50	98.32	97.16	97.03	93.57
Tree Size	29-71	25-47	21-25	19-29	15-25

Table 8. Experiment 1.4

Min	5	6	7*	8	9
CA	98.32	98.28	97.80	97.62	97.21
Tree Size	25-47	25-49	25-35	21-35	21-35

The highest CA achieved was 98.51. The tree is performing significantly better than chance and by a very large margin. This provides good evidence to reject the null hypothesis and accept that the tree is a good classifier. The baseline range of tree sizes (Conf = 0.25, MNO = 2) was 29-71 so the smallest were quite compact. Pruning achieved a modest improvement in the lower limit (25) and a good reduction in the upper limit (31) before a significant reduction in CA. So the overall conclusion is that the decision tree is performing very well the most straightforward (but also the most likely) form of classification it will be required to make.

Table 9 contains summaries of all of the experiments. The first row of data is the summary of experiment 1 (tables 5 – 8 shown above). Each of the following rows is the summary of a corresponding series of trials for a particular experiment.

Table 9. Best CA and pruning levels achieved for experiments in the series

Experiment	Baseline		Best Classifier		Best Pruned	
	%CA	Nodes	%CA	Nodes	%CA	Nodes
1	98.5	29-71	98.51	29-67	98.36	25-31
2	87.12	69-133	88.11	31-91	86.13	11-39
3	89.40	73-135	89.62	53-111	88.01	9-13
4	88.86	71-131	88.86	71-131	87.84	11-13
5	79.17	131-211	79.22	113-189	77.89	13-33
6	98.24	31-45	98.53	27-29	97.96	21-25
7	98.51	29-73	98.53	29-73	98.43	21-33
8	98.42	29-65	98.44	29-33	98.35	25-33
9	89.18	75-139	89.55	49-123	87.77	23-37
10	89.13	77-127	89.93	45-95	87.98	23-31
11	88.28	77-145	89.03	53-107	86.80	21-29
12	88.95	77-135	89.33	53-121	87.93	23-31
13	62.13	81-313	66.62	3-11	66.62	3-11†

4 Discussion of Results

All of the results are statistically significant improvements in CA over the chance level of 50%, providing evidence to accept all of the alternative hypotheses. Comparing the corresponding experiments the following observations may be made.

The best result obtained for the initial experiment, classifying straightforward questions in the presence of a mix of straightforward and difficult non-questions (in approximately equal proportions) was 89.43%. Experiment 1 (CA = 98.51) showed an increase in CA of 9.08% when discriminating between straightforward questions and straightforward non-questions. Experiment 9 (CA = 89.55) showed a slight improvement in CA of 0.12 over the initial experiment but a decrease in CA of 8.96% compared with experiment 1. The average increase in CA over the initial experiment obtained by using separate classifiers was 4.6%. This difference is statistically significant.

Experiment 5 shows that when the straightforward questions have a mix of 1, 2 and 3 word preambles in equal proportions applied, the CA decreases by 19.34% from experiment 1. However, the decreases in CA from experiment 1, when classifiers are trained for the specific preamble lengths, are 10.4%, 8.89% and 9.65% for experiments 2, 3 and 4 respectively. So there is an average improvement of the individual classifiers for preambles vs. the generic version of 9.69%. All of these differences are statistically significant.

Experiments 6, 7 and 8 correspond to experiments 2, 3, and 4 respectively. They were conducted to see if the preambles themselves were providing features that contributed to the CA of the classifiers in experiments 2-4. Adding the preambles to both classes resulted in the CAs reaching values very close to experiment 1, where no preambles were used. Effectively the average decrease of 9.69% has been

wiped out. The (statistically insignificant) differences were +0.02 for experiment 6, +0.02 for experiment 7 and -0.07 for experiment 8.

This is important because if the CA were significantly lower it would suggest that the preambles had contributed features assisting the classification process in experiments 2-4. In fact, the combined evidence from the experiments suggests that the preambles have an impact on classification accuracy, but that the classifiers are coping by learning to ignore them (to varying degrees). The results of experiments 6-8 suggest that although the preambles have an obscuring effect, they only add a little noise in terms of their own function word content.

Experiments 10, 11 and 12 combine the use of difficult non-questions with preambles applied to both questions and non-questions. When compared with the results for experiment 9, the differences were +0.38 for experiment 10, -0.52 for experiment 11 and -0.22 for experiment 12. Also the differences between experiments 10-12 and experiment 9 are only a little greater than the corresponding differences between experiments 6-8 and experiment 1, adding further weight to the inference that the preambles only add a little noise when inserted in front of both classes.

Pairwise comparisons of 10 with 6, 11 with 7 and 12 with 8 provide further insight on the impact of obscuring (or not obscuring) the first word feature in the presence of a preamble. Experiment 10 shows a decrease in CA of 8.6%, experiment 11 shows a decrease of 9.5% and experiment 12 shows a decrease of 9.11%. The average decrease is 9.07%, the decrease of 8.96 between experiment 1 and experiment 9 is very close to the average, in this case suggesting that adding the preambles has only made a small impact.

The final observations concern the results for experiment 13. The data for experiment 13 is created by concatenating two variable-length sentences. The closest equivalent is the use of mixed-length preambles in experiment 5.

The closest equivalent experiment is 5, which achieved a CA of 77.89%. At 66.62%, the CA of the simulated clauses was significantly lower. Although it exceeds chance performance by a statistically significant margin, it is still too low to be useful in a real-world classifier even after optimisation. Also, the optimisation process resulted in severe pruning of the DT classifiers generated. Consequently the final columns, marked with a †, repeat the values for the optimally pruned tree because it is not possible to prune it any further. Examining trees produced by this experiment shows they are dominated by a fairly simple split occurring at a word position approximately in the middle of the token string. This may simply be the best split that can be obtained when the discriminating features are smeared across the middle of the sentence by the concatenation process.

The decreases in CA for experiments 5 and 13 may be explicable by an increase in the complexity of the task, by dilution of the training data (1/3 as many training cases for each preamble in 5 and much worse in 13) or by a combination of the two. In any event, the individual classifiers clearly perform better than that for the mixed classes.

5 Conclusions and Future Work

The overall outcome of the experiments is encouraging. Four of the experiments produced classifiers with CAs of over 98% and one of theses categories represents the most likely question / non-question combination for dialogues where the participant follows Grice's rules. Of the remaining experiments, four produced classifiers scoring over 89% CA and 3 produced classifiers scoring over 88% CA. Only two of the experiments produced classifiers that had performance markedly below that required for use in real-world systems. In both these classifiers dilution of the training set may be a contributing factor in which case larger training sets may solve the problem. However, the final simulated clauses task may be too complex for this approach.

Future work may take a number of different directions. Before moving on to other dialogue acts it will be worthwhile developing the question classifiers further. The most obvious is an investigation of methods of combining the outputs of the current classifiers to produce a multi-classifier. This could involve a simple polling system or a more complex non-linear approach using Artificial Neural Networks for example. The second direction is to defer producing a multi-classifier until an investigation of optimisation of the specialised classifiers has been conducted, by optimising the feature extraction, for example. Finally, coping with sentences involving complex clauses with indirect and multiple dialogue acts may require a different method of presenting the tokenised sentences to a classifier. This could involve imposing a moving window with a limited size passing down the sentence from beginning to end, presenting each window position as a set of inputs to the classifier. This would also require a different training process.

Collectively, the experiments and directions for future work provide grounds to believe that Slim Function Word Classifiers can be improved to the point where they achieve CAs in excess of 95%.

References

1. Keizer, S.: A Bayesian Approach to Dialogue Act Classification. BI-DIALOG 2001 the 5th Workshop on Formal Semantics and Pragmatics of Dialogue, ZiF, Bielefeld (2001)
2. Webb, N., Hepple, M., Wilks, Y.: Dialogue Act Classification Based on Intra-Utterance Features. AAAI 2005. AAAI Press, Pittsburgh, Pennsylvania (2005)
3. Verbree, D., Rienks, R., Heylen, D.: Dialogue-Act Tagging Using Smart Feature Selection; Results On Multiple Corpora. IEEE Spoken Language Technology Workshop 70-73 (2006)
4. Venkataraman, A., Stolcke, A., Shriberg, E.: Automatic Dialog Act Labeling With Minimal Supervision. 9th Australian International Conference on Speech Science and Technology (2002)
5. Serafin, R., Di Eugenio, B., Glass, M.: Latent Semantic Analysis for dialogue act classification. The 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, Edmonton, Canada (2003)
6. Wermter, S., Lochel, M.: Learning Dialog Act Processing. COLING 1996, 16th International Conference on Computational Linguistics 740-745 (1996)
7. Searle, J.R.: Mind, Language and Society. Weidenfield & Nicholson (1999)
8. O'Shea, J., Bandar, Z., Crockett, K.: A Machine Learning Approach to Speech Act Classification Using Function Words. Lecture Notes in Artificial Intelligence vol. 6071, 82-91 (2010)
9. O'Shea, J., Bandar, Z., Crockett, K.: Using a Slim Function Word Classifier to Recognise Instruction Dialogue Acts. Lecture Notes In Artificial Intelligence vol. 6682, 26-34 (2011)
10. Längle, T., Lüth, T.C., Stopp, E., Herzog, G., G., K.: KANTRA — A Natural Language Interface for Intelligent Robots. In: Rembold, U., Dillman, R., Hertzberger, L.O., Kanade, T. (eds.): Intelligent Autonomous Systems (IAS 4), Amsterdam 357-364 (1995)
11. Crockett, K., Bandar, Z., O'Shea, J., McLean, D.: Bullying and Debt: Developing Novel Applications of Dialogue Systems. Knowledge and Reasoning in Practical Dialogue Systems (IJCAI). IJCAI, Pasadena, CA 1-9 (2009)
12. Biro, S., Hommel, B.: Becoming an intentional agent: Introduction to the special issue. Acta Psychologica vol. 124, 1–7 (2007)
13. Vytelingum, P., Voice, T.D., Ramchurn, S.D., Rogers, A., Jennings, N.R.: Intelligent Agents for the Smart Grid. The 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS '10), vol. 1, 1649-1650 (2010)
14. Bickmore, T., Giorgino, T.: Health dialog systems for patients and consumers. J Biomed Inform vol. 39, 556-571 (2006)
15. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by Latent Semantic Analysis. Journal of the American Society of Information Science vol. 41, 391-407 (1990)
16. Keizer, S., op den Akker, R., Nijholt, A.: Dialogue Act Recognition with Bayesian Networks for Dutch Dialogues. Third SIGdial Workshop on Discourse and Dialogue, Philadelphia 88-94 (2002)
17. van Rijsbergen, C.J.: Information Retrieval. Butterworths, Boston (1980)
18. Sanderson, M.: Stop Word List. `http://ftp.dcs.glasgow.ac.uk/idom/ir_resources/linguistic_utils/stop_words` (1994)
19. Spärck-Jones, K.: A Statistical Interpretation of Term Specificity and its Application in Retrieval. Journal of Documentation vol. 28, 11–21 (1972)
20. Salton, G., Wong, A., Yang , C.S.: A Vector Space Model for Automatic Indexing. Communications of the ACM vol.18, 613–620 (1975)
21. Deerwester, S., Dumais, S., Furnas, G.W., Harshman, R., Landauer, T., Lochbaum, K., Streeter, L.: Computer information retrieval using Latent Semantic Structure. In: Office, U.S.P. (ed.). Bell Communications Research Inc, United States of America (1989)
22. Bollacker, K.D., Lawrence, S., Giles, C.L.: CiteSeer: An Autonomous Web Agent for Automatic Retrieval and Identification of Interesting Publications. 2nd International ACM Conference on Autonomous Agents. ACM Press 116-123 (1998)
23. Li, Y., Bandar, Z., McLean, D., O'Shea, J.: Sentence Similarity Based on Semantic Nets and Corpus Statistics. IEEE Transactions on Knowledge and Data Engineering vol. 18, 1138-1150 (2006)
24. Islam, A., Inkpen, D.: Semantic Text Similarity using Corpus-Based Word Similarity and String Similarity. ACM Transactions on Knowledge Discovery from Data vol. 2, 1–25 (2008)
25. Quinlan, J.R.: C4.5: programs for machine learning. Morgan Kaufmann Publishers, San Mateo, California (1993)
26. Quinlan, J.R.: Induction of Decision Trees. Machine Learning vol. 1 81-106 (1986)
27. Lesch, S., Kleinbauer, T., Alexandersson, J.: A new Metric for the Evaluation of Dialog Act Classication. Dialor05, the Ninth Workshop On The Semantics And Pragmatics Of Dialogue (SEMDIAL 2005), Nancy, France (2005)
28. Kral, P., Cerisara, C., Kleckova, J.: Lexical Structure for Dialogue Act Recognition. Journal Of Multimedia vol. 2, 1-8 (2007)
29. Andernach, T., Poel, M., Salomons, E.: Finding Classes of Dialogue Utterances with Kohonen Networks. ECML/MLnetWorkshop on Empirical Learning of Natural Language Processing Tasks, Prague, Czech Republic 85–94 (1997)

30. Levin, L., Langley, C., Lavie, A., Gates, D., Wallace, D., Peterson, K.: Domain Specific Speech Acts for Spoken Language Translation. 4th SIGdial Workshop on Discourse and Dialogue, Sapporo, Japan (2003)
31. Clark, A., Popescu-Belis, A.: Multi-level Dialogue Act Tags. 5th SIGDIAL Workshop on Discourse and Dialog, SIGDIAL '04 Cambridge, MA. (2004)
32. Prasad, R., Walker, W.: Training a Dialogue Act Tagger For Human-Human and Human-Computer Travel Dialogues. The 3rd SIGdial workshop on Discourse and Dialogue, vol. 2 Philadelphia, Pennsylvania 162-173 (2002)
33. Webb, N., Hepple, M., Wilks, Y.: Error analysis of dialogue act classification. Proceedings of the 8th International Conference on Text, Speech and Dialogue, Karlovy Vary, Czech Republic (2005)
34. Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., Van Ess-dykema, C., Meteer, M.: Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech. Computational Linguistics vol. 26, 339-373 (2000)
35. Bui, T.H., Poel, M., Nijholt, A., Zwiers, J.: A tractable DDN-POMDP approach to affective dialogue modeling for general probabilistic frame-based dialogue systems. International Joint Conference on AI, IJCAI07 India, 2007 (2007)
36. Fernandez, R., Picard, R.W.: Dialog Act Classification from Prosodic Features Using Support Vector Machines. Speech Prosody (2002)
37. Jokinen, K., Hurtig, T., K., H., Kanto, K., Kaipainen, M., Kerminen, A.: Self-Organizing Dialogue Management. In: Isahara, H., Ma, Q. (eds.): The 2nd Workshop on Natural Language Processing and Neural Networks, NLPRS2001, Tokyo, Japan. 77–84 (2001)
38. Witten, I.H., Eibe, F.: Data Mining: Practical Machine Learning Tools and Techniques. Elsevier, San Francisco (2005)
39. Aleksander, I., Morton, H.: Introduction to Neural Computing. International Thomson Computer Press (1995)
40. Quirk, R., Greenbaum, S., Leech, G., Svartik, J.: A Comprehensive Grammar of the English Language. Addison Wesley Longman Ltd., Harlow (1985)
41. Flynn, R.: Question types - Glossary Definition - UsingEnglish.com. `http://www.usingenglish.com/glossary/question-types.html` (2002)
42. Christensen, C.R., Garvin, D.A.: Education for Judgment: The Artistry of Discussion Leadership. Harvard Business School Press (1992)
43. IRS: Frequently Asked Tax Questions and Answers. `http://www.irs.gov/faqs/index.html` (2009)
44. Saygin, A.P., Cicekli, I.: Pragmatics in human-computer conversations. Journal of Pragmatics vol. 34, 227-258 (2002)