Shallow Dialogue Processing Using Machine Learning Algorithms (or Not)

Andrei Popescu-Belis[1], Alexander Clark[1], Maria Georgescul[1], Denis Lalanne[2], and Sandrine Zufferey[1]

[1] University of Geneva, School of Translation and Interpreting (ETI), TIM/ISSCO, 40, bd. du Pont d'Arve, CH-1211 Geneva 4, Switzerland
andrei.popescu-belis@issco.unige.ch, asc@aclark.demon.co.uk, {maria.georgescul, sandrine.zufferey}@eti.unige.ch

[2] University of Fribourg, Faculty of Science, DIUF/DIVA, 3, ch. du Musée, CH-1700 Fribourg, Switzerland
denis.lalanne@unifr.ch

Abstract. This paper presents a shallow dialogue analysis model, aimed at human-human dialogues in the context of staff or business meetings. Four components of the model are defined, and several machine learning techniques are used to extract features from dialogue transcripts: maximum entropy classifiers for dialogue acts, latent semantic analysis for topic segmentation, or decision tree classifiers for discourse markers. A rule-based approach is proposed for solving cross-modal references to meeting documents. The methods are trained and evaluated thanks to a common data set and annotation format. The integration of the components into an automated shallow dialogue parser opens the way to multimodal meeting processing and retrieval applications.

1 Introduction

The design of computational methods to process dialogues between humans requires robust models and techniques for feature extraction. This paper proposes a shallow model for human dialogues that occur in meetings, along with a set of techniques for automatic detection of the features that constitute the model. The results of this *shallow dialogue analysis (SDA)* can be used in meeting processing and retrieval applications, to provide focussed accessed to the contents of the meetings [1].

The SDA approach to dialogue modelling has four major components, derived from state-of-the-art research in semantics and pragmatics: dialogue act tagging [2]; topic segmentation; detection of cross-modal references to documents [3]; and detection of discourse markers [4]. Machine learning algorithms are used to extract these features: their performances and their relevance to each task will be discussed.

S. Bengio and H. Bourlard (Eds.): MLMI 2004, LNCS 3361, pp. 277–290, 2005.

We will describe and justify the SDA model in Section 2, and provide also an annotation model and an outline of the available data. We analyse each of the four components of the SDA (cf. Sections 3, 4, 5 and 6) according to a common pattern: theoretical grounding; evaluation metric, available training data and reliability of ground truth annotation; machine learning techniques (or not) for automatic detection; performances of these techniques and discussion of their respective merits.

2 Shallow Dialogue Analysis

Modelling human dialogue is an active research area in linguistics and psychology, with applications to spoken and written language understanding by computers, and to human-computer interaction. There is, however, no generally accepted theory of human dialogue, and moreover, the various competing approaches to dialogue modelling are not quite tractable in terms of computational implementations. Our proposal of a shallow dialogue analysis model is inspired by some of the most commonly accepted hypotheses of dialogue theory, and driven by the needs of meeting processing and retrieval applications, while bearing in mind technical feasibility, robustness, and available resources.

2.1 Definition of SDA Model

Our model is composed of a set of features that combine information about the content or the state of the dialogue between two or more speakers. We consider that the dialogue unit is the *utterance*, a feature coded UT, i.e. a coherent unit of meaning that serves one function in the dialogue; the function is called *dialogue act* (DA) – another feature of our model. Although some theorists attempt to combine DA-labelled utterances into hierarchical structures, we observed that such structures are sometimes hard to annotate even for humans, and therefore do not consider them here as an SDA feature – hence the term 'shallow'.

We also consider a flat thematic structure: a dialogue is made of a series of disjoint episodes (EP), each of them dealing with a coherent topic. The extraction and resolution of references to entities is a key feature for all aspects of language understanding. We focus here on a particular type, namely the cross-modal references that are made by the speakers to documents and their sub-parts. The features are the boundaries of referring expressions (RE) as well as the links to the document elements they refer to (DE). Finally, we attempt to detect a particular class of words named *discourse markers* (DM) which play a particular role in dialogue: they can help the detection of the previous features, and can signal meta-linguistic properties of utterances, such as, here, hesitation or uncertainty.

2.2 Annotation Model

The annotation model for SDA presupposes the availability of spoken dialogue transcripts, in which the words uttered by each speaker are transcribed and

timed – the other required modality being the meeting documents. Such transcripts could be generated from separate channels (recorded using individual microphones) processed by an automated speech recognizer. However, an ASR system would have a word error rate of 30% or more in such an environment [5], especially since most "individual" microphone types are still sensitive to input from other speakers too. Therefore, manually corrected transcripts are preferable, done with Transcriber [6] and exported to XML format.

For an adequate representation of the SDA features, three types of annotations must be handled: boundaries, labels on bounded segments, and links from bounded segments to other elements. Given separate transcription files per channel, annotated as XML with time stamps, we annotate intra-channel boundaries (UT, RE, DM) on the transcription, and the other elements – cross-channel boundaries (EP), labels (DA, TO), and links (DE) – as separate XML elements, grouped into annotation blocks at the end of the files [7].

2.3 Annotated Data

Complete annotation of SDA from scratch is a time consuming task. Therefore, reuse of the existing resources summarized in Table 1 is a priority. Within the (IM)2 project, three main sites provide transcribed meeting recordings, with 4–8 participants: IDIAP, Univ. of Fribourg (UniFr), and ICSI.

Table 1. Available resources for SDA research

Institute	Nb. × time	Media	Lg.	Annotation
ICSI-MR	$75 \times 60'$	A,T	EN	UT,DA EP(30%),DM(60%)
IDIAP 1	$60 \times 5'$	A,V,T	EN	UT, EP
ISSCO 1	$8 \times 30'$	A,V	EN	ongoing
UniFr	$22 \times 15'$	A,V,T,D	FR	UT,RE,DE

The first two institutions provide transcripts and UT+EP annotation for ca. 60 and ca. 20 short meetings ($5'$-$15'$), and a larger corpus is currently being recorded at IDIAP [8]. These resources consist of multimodal data (audio, video and transcription). UniFr also provides meeting documents, therefore we annotated this data with references to documents (RE, DE). The ICSI-MR project has about 75 one-hour meetings annotated with UT, DA [9], which we validated and converted to SDA format [2]. Annotation of EP boundaries on 25 ICSI-MR meetings was available from another source [10]. A series of meetings was recorded by ISSCO at IDIAP (spring 2004) and is currently being transcribed and annotated.

Stylesheets were written and conversion methods were defined for these resources, which await complete annotation of the missing SDA features. The training and test data used below makes use of all the available annotations.

3 Dialogue Acts

3.1 Dialogue Act Tagsets

An utterance is a coherent, contiguous series of words from a given speaker, which serves a precise function in the dialog. An utterance can often be equated with a proposition or a sentence, but in spoken language, utterances do not always correspond to well-formed or completed propositions. In this section, using ICSI-MR pre-segmented data (UT annotation), we will focus on the automatic assignment of dialogue functions to utterances, that is, *dialogue acts* (DA annotation).

There is little consensus on a set of DAs, since tagsets depend on the goals of their creators [11]. Among the many existing DA tagsets, the multidimensional DAMSL [12] and the one-dimensional SWBD-DAMSL [13] were used to label two-party conversations. While DAMSL offers about 4 million tag combinations, SWBD-DAMSL retains only the most frequent ones, i.e. 42 mutually exclusive tags such as 'statement', 'opinion', 'agree/accept'. SWBD-DAMSL is well adapted to automatic DA annotation and was used for language modelling in speech recognition [14].

The ICSI-MR tagset [15], used for the ICSI-MR data, extends SWBD-DAMSL, and allows one utterance to be marked with as many tags as needed. Our formalization of the ICSI-MR tagset using rewriting rules shows that the number of possible combinations of tags (DA labels) reaches several millions, which makes a huge search space for automatic DA tagging [2].

3.2 The MALTUS DA Tagset

We defined MALTUS (Multidimensional Abstract Layered Tagset for Utterances) in order to reduce the search space, by assigning exclusiveness constraints among tags, while remaining compatible with ICSI-MR. MALTUS is more abstract than ICSI-MR, but can be refined. An utterance is either marked U (undecipherable) or it has a general tag followed by zero or more specific tags. It can also bear a disruption mark. More formally:

DA → (U | (gen_tag (spc_tag)?)) (.D)?
gen_tag → S | Q | B | H
spc_tag → (RP | RN | RU)? RI? AT? DO? PO?

The glosses of the tags, generally inspired from ICSI-MR, are: U undecipherable, S statement, Q question, B backchannel, H hold, RP/RN/RU positive/negative/other answer, RI restated information, DO command or other performative, AT attention management (acknowledgement, tag question, etc.), PO politeness (apology, thanks, etc.), D disruption (interrupted, abandoned). There are only about 500 possible MALTUS labels (combinations of tags), but observations of the converted ICSI-MR data show that their distribution is very skewed; for instance, about 75% of the labels contain a S tag.

3.3 Automatic DA Annotation

In the experiments we present here, we focus on the multi-dimensional nature of the MALTUS tagsets, and explore the extent to which such a tagset can be predicted by classifying each dimension separately – i.e. by having a set of "orthogonal" classifiers – as opposed to classifying the entire structured object in a single step using a single multi-class classifier on a flattened representation. In prior research, some form of sequential inference algorithm has been used to combine the local decisions about the DA of each utterance into a classification of the whole utterance. The common way of doing this has been to use a hidden Markov model to model the sequence and to use a standard decoding algorithm to find either the complete sequence with maximum a posteriori (MAP) likelihood or to select for each utterance the DA with MAP likelihood. Here, we will ignore this complexity and allow our classifier access to the gold standard tags of the previous utterances - making the preliminary task substantially easier.

Since for the moment we are not using prosodic or acoustic information, but just the dialogue transcriptions, there are two sources of information that can be used to classify utterances with respect to dialogue acts: first, the sequence of words that constitutes the utterance, and second, the surrounding utterances and their classification. Hence, two sorts of features will be used here: internal lexical features derived from the words in the utterance, and contextual features derived from the surrounding utterances. We used as lexical features the 1000 most frequent words, together with additional features for these words occurring at the beginning or end of the utterance. This gives an upper bound of 3000 lexical features. We used some simple contextual features relating to basic temporal relationships between adjacent utterances such as precedence and overlap.

3.4 Results

We use a Maximum Entropy (ME) classifier which allows an efficient combination of many overlapping features. We selected 5 ICSI-MR meetings (6771 utterances) to use as our test set and 40 as our training set, leaving the others for possible later experiments. As a simple baseline we use the classifier which just guesses the most likely DA tag (S). We first performed some experiments on the original ICSI-MR tagset, to see how predictable it is. We defined a simple six-way classification task which classifies disruption forms, undecipherable forms, and the four general tags S, Q, B, H mentioned above. This is an empirically well-founded distinction: the ICSI-MR group reported inter-annotator agreement of $\kappa = 0.79$ (using the *kappa* measure [16]) for a very similar task. Our ME classifier scored 77.9% accuracy, against a baseline of 54.0%. A more relevant performance criterion for our application is the accuracy of classification into the four general tags S, Q, B, H. In this case we removed disrupted and undecipherable utterances, slightly reducing the size of the test set, and achieved a score of 84.9% (baseline 64.1%).

With regard to the MALTUS tagset, since it has some internal structure, it should accordingly be possible to identify the different parts separately, and

then combine the results. We have therefore performed some preliminary experiments with classifiers that classify each level separately. We again removed the disruption tags since in our current framework we are unable to predict them accurately. The baseline for this task is again a classifier that chooses the most likely tag (S) which gives 41.9% accuracy. Using a single classifier on this complex task gave an accuracy of 73.2%.

We also trained six separate classifiers and combined the results. This complex classifier gave an accuracy of 70.5%. This mild decrease in performance is rather surprising – one would expect the performance to increase as the data sets for each distinction get larger. This can be explained by non-trivial dependencies between the classifications. There are a number of ways this could be treated, using either structured output spaces or stacked classifiers, where each classifier can use the output of the previous classifier as a feature in the next one. It is also possible that these dependencies reflect idiosyncrasies of the tagging process: tendencies of the annotators to favour or avoid certain combinations of tags. We expect the performance of a final, fully automatic classifier to be substantially higher than the results presented here, owing to the use of more powerful classifiers and, more importantly, larger and richer feature sets.

4 Topic Segmentation

4.1 Definition and Input Data

Segmentation into thematic episodes – defined as units which tend to reflect coherence around particular topics – plays an important role in automatic summarization, or in meeting indexing and retrieval. We aim here at finding the most prominent boundaries between episodes, without building a hierarchic topical structure of each meeting, hence making minimal theoretical assumptions about discourse structure. Previous studies of automatic thematic segmentation were based on various (probabilistic) lexical cohesion methods, or combined multiple features such as cue phrases and prosodic features. Their application to multi-party dialogues, as opposed to narrative or descriptive texts, remains less explored.

While focusing on multi-party dialogues, we also use narrative texts for setup and comparison purposes. We use three sets of test data, of similar length, ca. 90,000 words, without stopwords. For the ICSI-MR dialogue data, the topic boundaries for 25 meetings were defined by the consensus of at least three annotators [10]. There is an average of 7.32 episodes per one-hour meeting (test sample). Cochran's Q test showed that annotation reliability is significant at a 0.05 level. The TDT3 collection of news stories has an average of 24 segments per test sample (one news report). The subset of the Brown corpus is an artificial test set [17], where each test sample consists of ten text segments (topics). A segment contains the first n sentences ($3 \leq n \leq 11$) of a randomly selected document from the Brown corpus.

4.2 Methods for Automatic Segmentation

We investigated an approach based on Latent Semantic Analysis (LSA). LSA is generally used to induce and to represent aspects of the meaning of words reflected in their natural language usage [18], and we describe below its application to SDA annotation – first the training phase, then the test phase.

During the *learning phase*, we consider that available data is tagged at the thematic episodes level. Hence, we have the following types of segments (blocks) from the input data: a human-annotated topic segment for the ICSI-MR data; a story unit for the TDT and Brown data.

The segmented input data is first filtered to remove the most common words. Then each input block is represented in a vector space model as a n-dimensional vector, where n is the number of distinct terms in the vocabulary. So, a_{ij}, the ith element of the jth vector, is a function of the frequency of the ith vocabulary term in the corresponding block of text. Using results from information retrieval [19], this function of the frequency is expressed as: $a_{ij} = l_{ij} \cdot g_i$, where l_{ij} and g_i are local and global weights respectively. As local weightings we use: *Term Frequency (TF)*, *Binary* and *Log*. The global term weighting functions that we used are: *Normal*, *GfIdf*, *Idf*, and *Entropy*. The data matrix $A_{n\times m} = (a_{ij})_{1\leq i\leq n, 1\leq j\leq m}$, where m is the number of blocks of text, is called the matrix of frequencies.

After the construction of the matrix of frequencies, the method projects this matrix into an appropriate lower space dimension. This is done by performing a rank-k approximation to $A_{n\times m}$ by using its singular value decomposition [20]: $A_{n\times m} \approx \hat{A}_{n\times m} = U_{n\times k} \cdot \Sigma_{k\times k} \cdot V^T_{m\times k}$, where $k \ll \min(n, m)$ is the order of the decomposition, ^{T} denotes matrix transposition, and the diagonal matrix $\Sigma_{k\times k}$ contains the first k singular values of A in descending order. The idea behind this equation is that terms that are semantically associated are placed to some degree near one another in the subspace representation, i.e. some words that have similar co-occurrence patterns are projected into the same dimension. Finding the optimal dimensionality of the LSA reduced space is an empirical issue.

In the *test phase* of LSA, we compute the proximity between the utterances of a test sample. Given a text to be segmented, the representation of each utterance $\hat{u}$ is computed using the equation $\hat{u} = u \cdot U_{n\times k} \cdot \Sigma^{-1}_{k\times k}$ (the version without $\Sigma^{-1}_{k\times k}$ was also tested), where $U_{n\times k}$ and $\Sigma_{k\times k}$ were determined in the training phase; $u = (u_1, u_2, \ldots, u_n)$ with u_i indicating the weighted frequency of the ith vocabulary term in the utterance (using the same local and global weighting functions applied in the learning phase). The thematic distance between two utterances is then computed using the cosine metric, and topic boundaries are identified by a divisive clustering procedure [17].

4.3 Results

We evaluated mainly the relations between different factors that influence the results obtained by LSA, such as frequency matrix transformations, choice of the reduced LSA space dimensionality, choice of applying or not a ranking function

on the similarity matrix before clustering. Singular value decomposition was performed using the single-vector Lanczos method implemented by Berry [20]. The baseline scores of the following simplistic algorithms are used for comparison: (1) *ALL*: considers all potential boundaries as real boundaries; (2) *NONE*: no boundary at all; (3) *RANDOM*: randomly select the boundaries. We have also experimented with the state-of-the-art algorithm developed by Choi [17], labelled C99. All the algorithms are given the correct number of segments (boundaries) in the test texts. We use the P_k error metric for evaluation [21]: P_k is the probability that a randomly chosen pair of words from a window of dimension k (k being the mean length of an episode in the reference data) is wrongly classified as being in the same segment or not.

Our findings show that the C99 algorithm has only 9% error rate on the subset of Brown corpus, but its performance decreases at 23% error rate on TDT data, and attains 37% error rate on ICSI-MR – which is even bigger than the error rate of 33.10% given by the baseline algorithm *NONE*.

In our preliminary experiments on ICSI-MR data we trained the LSA model on a dataset containing 6,124 terms. The LSA algorithm gives an error rate of about 35% when no ranking is applied. Thus the LSA results are slightly better than the C99 results, but the error rates are still higher than those given by the baseline algorithm *NONE*.

Our experiments on TDT data were done by training the LSA model on a dataset containing 63,667 terms. The error rates obtained are about 36% and we observe a slight improvement in the LSA performance (at 34%) when $Log \cdot Entropy$ was adopted instead of $TF \cdot Idf$ as initial term weighting. However, C99 performs better than LSA on TDT data. Besides, we obtained an error rate of 34.14% on the Brown data, when training was performed on a Brown subset containing 6,498 terms after the pre-processing step.

Depending on the training data, it appears that LSA applied to topic segmentation does not perform better than other, less time-consuming approaches such as C99. Our experiments show that for topic segmentation, if we interpret LSA as a mechanism for representing the terms of the collection, this technique alone is insufficient for dealing with the variability in term occurrence.

5 References to Documents

5.1 Definition of the Component

The detection of references made by the speakers to the meeting documents is an SDA component that contributes to the general understanding of a dialogue, and is related to another communication modality, namely documents (agenda, reports, memos, notes, slides, etc.). We deal here with press-review meetings that discuss the front pages of one or more newspapers.

The task requires (a) the detection of the referring expressions (REs) that make reference to the documents of the meeting, and (b) the attachment of each RE to the document element it refers to. We focus here on task (b), using REs

identified by humans. Task (a) could be carried out using a repertoire of pattern matching rules.

Newspaper front pages have a hierarchical structure made of elements that can contain other elements – hence a straightforward encoding in XML. For instance, a `Newspaper` front page bears the newspaper's `Name` and `Date`, one `Master Article`, one or more `Article`s, etc. For simplicity of annotation, each content element has an `ID` attribute bearing a unique index. Inferring the structure of a document from its graphical aspect encoded in PDF is a task that can be automated with good performances [22]. In what follows, we use manually-generated XML representations of documents, considered 100% accurate.

In summary, the annotation task requires the construction of the correct pointers from the RE indexes to the document names and document elements, which are characterized by ID or by XPath.

5.2 Evaluation Method and Data

For evaluation, one must compare for each RE the referent (document element) found by the system with the correct one selected by the annotators. If the two are the same, the system scores 1, otherwise it scores 0. The total score is the number of correctly solved REs out of the total number of REs. The automatic evaluation measure we implemented provides two scores: (1) the number of times the document is correctly identified, and (2) the number of times the document element, characterized by its `ID` attribute, is correctly identified.

The annotation of the gold standard was done for 15 UniFr meetings with a total of 322 REs referring to documents, and 1 to 4 documents per meeting. Inter-annotator agreement, measured on 3 meetings with 92 REs, reaches 96% for document assignment (3 errors), and 90% on document elements (9 errors). After discussion among annotators, 100% agreement was reached on document assignment, and 97% agreement on document elements – both very high scores.

5.3 Ref2doc Algorithm Based on Anaphora Tracking

Although machine learning can be applied to coreference resolution, the scarcity of data with respect to the variety of features needed to assign a referent to an RE prompted us to define a rule-based algorithm which exploits the distinction between anaphoric and non-anaphoric REs, and the co-occurrences of words between the RE (plus context) and each document element.

The algorithm scans each meeting transcript and stores as variables the 'current document' and the 'current document element' (or article). For each RE, the algorithm determines first the document it refers to, from the list of documents associated to the meeting. REs that make use of a newspaper's name are considered to refer to the respective newspaper; the other ones are supposed to refer to the current newspaper, i.e. they are anaphors.

The algorithm then attempts to assign a document element to the current RE. First, it attempts to find out whether the RE is anaphoric or not, by matching it against a list of typical anaphors: 'it', 'the article', 'this article', 'the author'.

If the RE is anaphoric, then it is associated to the current article or document element (except for the first one, which is never anaphoric).

If the RE is not considered to be anaphoric, then the algorithm attempts to link it to a document element by comparing the content words of the RE with those of each article. The words of the RE are considered, as well as those in its left and right contexts. A match with the title of the article, or the author name, is weighted more than one with the content. Finally, the article that scores the most matches is considered to be the referent of the RE, and becomes the current document element.

5.4 Results and Observations

The baseline score for RE ↔ document association, obtained when always choosing the most frequent newspaper, is 82% accuracy (265 REs out of 322). But some meetings deal only with one document; if we look only at meetings that involve at least two newspapers, then the baseline score is 50% (46/92), a much lower value. Regarding RE ↔ document element association, if the referent is always the front page as a whole, then accuracy is 16%. If the referent is always the main article, then accuracy is 18%.

Our algorithm reaches 98% accuracy for the identification of documents referred to by REs, or 93% if we take into account only the meetings with several documents.

The accuracy for document element identification is 73% (237 REs out of 322). If we count only REs for which the document was correctly identified, the accuracy is 74% (236 REs out of 316). This score is obtained when only the right context of the RE is considered (i.e. the words after the RE), not the left one. Also, the optimal number of words to look for in the right context is about ten. Without the right context, the score drops at 40%. Finally, if anaphor tracking is disabled, the score drops at 65%, which shows the relevance of this feature.

At current levels of performance, the resolution of references to documents appears to be an efficient cross-channel process that enhances dialogue and document processing, and helps the multi-media rendering of the results.

6 Discourse Markers

6.1 Definition and Evaluation

The identification of discourse markers (DMs) – words like *actually*, *but*, *I mean*, *like*, *well* – is relevant to lower-level analysis processes such as POS tagging, parsing, or to SDA components such as DA tagging. From an SDA point of view, the detection of *like* as a DM is useful to indicate approximation, uncertainty, or fuzziness in a dialogue. As for the DM *well*, it can be used to detect topic shifts.

The present SDA component disambiguates occurrences of two important DMs, *like* and *well*, that is, separates the occurrences when they function as DMs ("pragmatic uses") from their other occurrences. For instance, *like* can be

used as a preposition, adjective, conjunction, adverb, noun, verb – or as a DM, as in this example from ICSI-MR: "It took *like* twenty minutes".

The *kappa* metric [16] can be used to compare human annotations, or to score a system against a gold standard. A simpler but useful metric here is the percentage of occurrences correctly identified, or accuracy. We annotated all occurrences of *like* and *well* as DMs in 50 one-hour ICSI-MR dialogues, finding about 800 and 600 occurrences of each. When all the occurrences are classified as DMs we obtain a baseline accuracy of 37%, resp. 66%. Inter-annotator agreement reaches $\kappa = 0.65$ for the identification of the DM *like*, provided the audio is available, for prosodic cues [4]. Furthermore, to evaluate the retrieval of pragmatic uses among all uses, recall and precision are also relevant.

6.2 Automatic Detection of DMs

Three methods were tested for the detection of *like* as a DM: a simple rule-based filter, a part-of-speech tagger, and a decision-tree classifier trained on the available data. The last method, which provided the best results, was then applied to *well*.

Using first a list of collocations in order to filter out occurrences which are not DMs (e.g. *I like*, or *looks like*) we score 0.75 precision with 100% recall. A significant number of non pragmatic occurrences are thus correctly ruled out using quite a simple filter. Besides, none of the pragmatic uses was missed in the process.

Experiments with QTag, a probabilistic part-of-speech tagger [23], investigated whether the DMs could be disambiguated using POS tags, by filtering out the non-pragmatic uses, such as the cases when *like* is a verb and *well* an adverb. For the occurrences of *like*, QTag assigns mostly 'preposition' (1,412 occ.) and 'verb' (509 occ.) tags. When 'verb' is used to filter out non-DMs, recall is 0.77, precision is 0.38, accuracy 44%, and κ is only 0.02. Other interpretations of the tags do not lead to better results. The main reason that explains the failure of the tagger to detect DM uses of *like* is that it was trained only on written material.

Finally, we used the C4.5 decision tree learner (WEKA toolkit [24]) with 10-fold cross-validation of classifiers. For each occurrence of *like*, the following features were extracted automatically: (1) presence of a collocation that rules out the presence of a DM; (2) duration of the spoken word *like*; (3) duration of the pause before *like* (or initial *like*); (4) duration of the pause after *like* (or final *like*).

The best performance obtained by a C4.5 classifier is 0.95 recall and 0.68 precision for identifying DM occurrences of *like*, corresponding to 81% correctly classified instances and $\kappa = 0.63$. This is a significant performance, but it appears to be in the same range as the filter-based method. Indeed, the decision tree exhibits as the first nodes the two classes of collocation filters, thus offering a strong empirical proof of their relevance. The next feature in the tree is the duration of the pause before *like*: a relatively long pause before *like* characterizes

a DM. The next features in the tree have quite a low precision, and may not generalize to other corpora.

The best classifier tends to show that apart from the collocation filters, the other features do not play an important role. Indeed, a classifier based only on the collocation filters achieves 0.96 recall and 0.67 precision for DM identification (80% correctly classified instances and $\kappa = 0.62$), which is only slightly below the best classifier. Is it that the time-based features are totally irrelevant? An experiment without the two collocation filters shows that temporal features *are* relevant, since the best classifier achieves 67% correct classification ($\kappa = 0.23$); but they are superseded by collocation-based features, when available.

The features defined for *well* are similar to those used for like: collocation-based filters and time-based features. The highest classification accuracy after training, 91% and $\kappa = 0.8$, is obtained by a decision tree combining the collocation filters and the duration of the pause after *well*. This corresponds to 91% precision and 97% recall for the DM detection task. Here again the collocation-based features provide the best classification but other time-based features alone also perform above chance.

For both DMs, the results suggest that time-based features could generalize to a whole class of DMs, though for individual DMs, such features are outperformed by collocations filters based on patterns of occurrences. Given the strong pragmatic function of DMs, it is unlikely that low-level features combined with machine learning will entirely solve the problem. However, even a partial classification could help improve SDA.

7 Conclusion and Perspectives on SDA

This paper has shown how a variety of machine learning techniques can be used to detect a set of features in dialogue transcripts. Three of the four shallow components of our model – dialogue acts, discourse markers of uncertainty, and topic segmentation – can be reliably learned from training data using statistical techniques. Two of these components are based on classifiers, but the instances to be classified are independent for DMs, and correlated for DAs. Topic segmentation requires, in a certain sense, the classification of sets of words, hence it makes use of different techniques. For one component – references to documents – machine learning did not appear, at this stage, to provide a tractable solution, since the correspondence between REs and DEs was better modelled by a set of hand-written rules. The various techniques are "shallow" as they do not build complex dialogue structures, and they process the dialogue flow quite linearly. They do not make use of complex linguistic knowledge, but of robust low-level features. Future work will study the possibility to extend the available linguistic resources without reducing coverage too much.

The components of the SDA model are in fact interdependent, which could allow for an integrated annotation mechanism. For instance, consecutive references to the same article often correspond to an episode, or episode boundaries are related to some types of dialogue acts. These relations must first be studied

empirically, on manually annotated data. Then, components can be integrated using the following blackboard-style mechanism. Components can add annotations to the XML data, depending on existing annotations, but not to delete or change them, to avoid infinite loops. The SDA parser executes consecutively each component, which checks if the annotation has changed since it last processed it; if it has, then the component reprocesses the data, possibly adding new annotations. The process stops when no component is able to add new annotations.

New components should also be added to the SDA parser, based on ongoing studies of user needs and on tractability. Attention will be paid to annotations derived from other modalities, such as the use of facial expression for DA annotation.

The SDA annotations are the main features that are stored in a database of meetings, to allow meeting retrieval and browsing. Users can submit queries based on the SDA component features, to retrieve utterances from a dialogue and their context. The SDA annotations enable the production of a rich transcript of the meeting, which can be used for browsing, as a master modality that gives access to other modalities (e.g. audio and video), and in particular to the relevant meeting documents.

References

1. Armstrong, S., Clark, A., Coray, G., Georgescul, M., Pallotta, V., Popescu-Belis, A., Portabella, D., Rajman, M., Starlander, M.: Natural language queries on natural language data: a database of meeting dialogues. In: NLDB '03, Burg, Germany (2003) 14–27
2. Clark, A., Popescu-Belis, A.: Multi-level dialogue act tags. In: SIGDial '04, Cambridge, MA (2004) 163–170
3. Popescu-Belis, A., Lalanne, D.: Reference resolution over a restricted domain: References to documents. In: ACL'04 Workshop on Reference Resolution and its Applications, Barcelona (2004) 71–78
4. Zufferey, S., Popescu-Belis, A.: Towards automatic identification of discourse markers in dialogs: The case of like. In: SIGDial '04, Cambridge, MA (2004) 63–71
5. Morgan, N., Baron, D., Bhagat, S., Carvey, H., Dhillon, R., Edwards, J.A., Gelbart, D., Janin, A., Krupski, A., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A., Wooters, C.: Meetings about meetings: research at ICSI on speech in multiparty conversations. In: ICASSP '03, Hong Kong, China (2003)
6. Barras, C., Geoffrois, E., Wu, Z., Liberman, M.: Transcriber: development and use of a tool for assisting speech corpora production. Speech Comm. **33** (2001) 5–22
7. Popescu-Belis, A., Georgescul, M., Clark, A., Armstrong, S.: Building and using a corpus of shallow dialogue annotated meetings. In: LREC 2004, Lisbon, Portugal (2004) 1451–1454
8. McCowan, I., Bengio, S., Gatica-Perez, D., Lathoud, G., Monay, F., Moore, D., Wellner, P., Bourlard, H.: Modeling human interaction in meetings. In: ICASSP 2003, Hong Kong, China (2003)
9. Shriberg, E., Dhillon, R., Bhagat, S., Ang, J., Carvey, H.: The ICSI meeting recorder dialog act (MRDA) corpus. In: SIGDial '04, Cambridge, MA (2004) 97–100

10. Galley, M., McKeown, K., Fosler-Lussier, E., Jing, H.: Discourse segmentation of multi-party conversation. In: ACL '03, Sapporo, Japan (2003) 562–569
11. Traum, D.R.: 20 questions for dialogue act taxonomies. Journal of Semantics **17** (2000) 7–30
12. Allen, J.F., Core, M.G.: DAMSL: Dialog act markup in several layers. Technical Report draft 2.1, Multiparty Discourse Group, Discourse Research Initiative (1997)
13. Jurafsky, D., Shriberg, E., Biasca, D.: Switchboard SWBD-DAMSL shallow discourse function annotation: Coders manual. Technical Report 97-02, draft 13, Univ. of Colorado, Inst. of Cognitive Science (1997)
14. Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., Meteer, M., Van Ess-Dykema, C.: Dialogue act modeling for automatic tagging and recognition of conversational speech. Comp. Ling. **26** (2000) 339–371
15. Dhillon, R., Bhagat, S., Carvey, H., Shriberg, E.: Meeting recorder project: Dialog act labeling guide. Technical Report TR-04-002, ICSI, Berkeley, CA (2004)
16. Carletta, J., Isard, A., Isard, S., Kowtko, J.C., Doherty-Sneddon, G., Anderson, A.H.: The reliability of a dialogue structure coding scheme. Comp. Ling. **23** (1997) 13–31
17. Choi, F.Y.Y.: Advances in domain independent linear text segmentation. In: NAACL 2000, Seattle, WA, USA (2000) 26–33
18. Bellegarda, J.R.: Exploiting latent semantic information in statistical language modeling. In: Proceedings of the IEEE. Volume 88. (2000) 1279–1296
19. Dumais, S.: Improving the retrieval of information from external sources. Behavior Research Methods, Instruments and Computers **23** (1991) 229–236
20. Berry, M.W.: Large scale singular value computations. International Journal of Supercomputer Applications **6** (1992) 13–49
21. Beeferman, D., Berger, A., Lafferty, J.: Statistical models for text segmentation. Mach. Learn. **34** (1999) 177–210
22. Hadjar, K., Rigamonti, M., Lalanne, D., Ingold, R.: Xed: a new tool for extracting hidden structures from electronic documents. In: Workshop on Document Image Analysis for Libraries, Palo Alto, CA (2004)
23. Mason, O.: Programming for Corpus Linguistics: How to do Text Analysis in Java. Edinburgh University Press, Edinburgh, UK (2000)
24. Witten, I., Frank, E.: Data Mining: Practical Machine Learning Tools with Java Implementations. Morgan Kaufmann, San Francisco, CA (2000)