Determining Latency for On-line Dialog Act Classification

Sebastian Germesin, Tilman Becker, and Peter Poller

Deutsches **F**orschungszentrum für **K**ünstliche **I**ntelligenz GmbH
Stuhlsatzenhausweg 3, 66123 Saarbrücken, Germany
firstname.lastname@dfki.de

Abstract. This paper presents results from our ongoing research on the recursive classification of dialog acts. We successfully used dynamic features like the label of previous and future dialog acts as features in a statistical machine learning approach to gain information about the class of the current dialog act. Using these features in a real-time application system leads to the problem that the labels of the upcoming dialog acts are not available when classifying the current one. Thus, these features change over time and when new dialog acts get classified, the already classified dialog acts have to be re-classified with the new information. We found that a latency of about 60 dialog acts which corresponds to nearly 2 minutes is sufficient to reach the maximized detection rate. Furthermore, a latency of already 30 segments (60 seconds) yields an improvement of about 50% of the maximum achievable improvement.

1 Introduction

Dialog Act (DA) segmentation and classification of speech is an important aspect for several subsequent processing steps like, e.g., discourse modeling or topic descriptions. A variety of methods have been developed and used for the classification task (e.g., [1], [2] or [3]) and it is well known that words and phrases in DAs are the strongest cues to their identification [4]. Furthermore, there exist many other features that help to increase the performance of DA classification systems: *prosodic cues, speaker-related features* and *dynamic relations*. One major problem is the transformation of these features from the development system to an on-line application where many distorting circumstances prevent their smooth estimation.

Our approach aims to improve the usage of the *dynamic* features that describe the relationship between the current DA to its neighbors. These features change over time when new elements get classified and hence leads to a re-classification of the DAs.

We currently examine the DA scheme on the AMI Meeting corpus which is is explained in detail in sections 2 and 3. AMI (Augmented Multi-party Interaction) is a multi-disciplinary research project to "develop technology to support human interaction in meetings and to provide better structure in the way meetings are run and documented" [5].

2 AMI Corpus

The AMI project published a speech corpus with more than 100 hours of four person project meetings [5]. These meetings are all held in English and the task of the particular participants is to design a television remote control. Next to the transcribed speech of the participants, the corpus offers different *annotation layers* that contain a variety of information (e.g., extractive summaries, ASR output or topics). Additionally, all meetings are annotated with the dialog act scheme that is presented in the following section. The separation into training, development and test set has been fixed to ensure easily comparisons (see table 1).

Subset	Meetings	#meetings	#series
Training set	ES2002, ES2005-2010, ES2012-2016, IS1000-1007, TS3005, TS3008-3012	98	25
Development set	ES2003, ES2011, IS1008, TS3004, TS3006	20	5
Evaluation set	ES2004, ES2014, IS1009, TS3003, TS3007	20	5

Table 1. Split of the AMI data into training, development and test set (from [6])

3 Dialog Act Scheme

Dialog Acts are labels for the utterances which roughly categorize the speaker's intention and hence, can be used in various ways. The AMI dialog act tag set consists of 15 dialog act types which are organized in 6 major groups (see table 2). We do not want to explain all classes in detail as this would exceed the scope of this study but we refer to the corresponding annotation manual[1] where each class is explained in detail.

Information exchange Giving and eliciting information
Possible actions Making or eliciting suggestions or offers
Comments Making or eliciting assessments and comments about understanding
Social acts Expressing positive or negative feelings towards individuals or the group
Backchannel, Stall and Fragment Classes for utterances without content, which allow complete segmentation of the material
Other A remainder class for utterances which convey an intention, but do not fit into the five previous categories

[1] http://mmm.idiap.ch/private/ami/annotation/dialogue_acts_manual_1.0.pdf

DA category	Dialog Acts
Info exchange	Inform, Elicit Inform
Actions	Suggest, Offer, Elicit-Offer-Or-Suggestion
Comments	Assess, Comment-About-Understanding Elicit-Assessment, Elicit-Comment-About-Understanding
Social acts	Be-Positive, Be-Negative
(Segmentation)	Backchannel, Stall, Fragment
Everything else	Other

Table 2. Overview of Dialog Act Scheme

4 System

The machine learning classifier is implemented with the help of the freely available WEKA toolkit [7] which contains many state-of-the-art machine learning algorithms and a variety of evaluation metrics. Furthermore, it allows to adapt other algorithms due to its simple interface. In fact, we added an implementation of the Maximum Entropy classification algorithm, by the Stanford NLP group[2] to the WEKA library and used it in this study.

The system is designed to use recursive classification for the dialog act labeling process. This means that it (pre-)classifies the current DA without any information about the future DA labels. If new DAs get segmented and classified, the system starts to re-classify the previous DAs which now have updated information about their future DAs. Furthermore, if any labels of the re-classified DAs get updated, the number of updating DAs gets increased to ensure that the new information gets propagated backwards to all DAs.

1. step: DA_i gets segmented
2. step: Classification of DA_i
3. step: Re-Classification of previous j DAs ($DA_{i-1-j}...DA_{i-1}$)
4. step: If Label of DA_k changes ($k \leq i-1$), $j \leftarrow j+i-k$
5. step: Until: stable labeling

Figure 1 visualizes this back-propagation where the sixth DA is currently being classified and leads to an updating of all previous DAs. Now, the label-change of the third DA leads to an update of the fourth label which updates the fifth DA and henceforth has an impact on the first DA. As this design could possibly lead to a so-called livelock[3], we had to avoid this. Hence, we limited the amount of updatings per new classified segments.

[2] http://http://nlp.stanford.edu/software/classifier.shtml

[3] A livelock is similar to a deadlock, except that the states of the processes involved in the livelock constantly change but none is progressing.

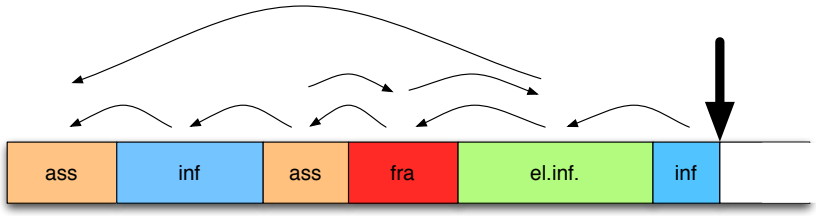

Fig. 1. Sketch of the system's back-propagation

5 Experimental Results

This section describes the results that we gained on the evaluation of the online dialog act classification. The presented results are estimated over meeting ES2004a which consists of 461 dialog act segments and a real-time of 1048 seconds (about 17 minutes). We decided that this suffices for the evaluation of our approach. Nevertheless, the system is designed to label several meetings in a batch process but is also able to work as a real-time application in an ongoing meeting.

The system's latency baseline is limited to the duration of one dialog act added to the duration of the following word as this has to be classified as a start of the next DA. This corresponds to about 2 seconds as this is the average duration of a DA. Hence, we can state that after 2 seconds, we get a first label of the DA with an accuracy of 54.88% (see table 3).

history	**accuracy [%]**	**worst latency [s]**	**total time [s]**
0	54.88	2	013
10	56.83	76	141
20	57.27	108	211
40	58.78	128	409
80	58.78	272	754

Table 3. Results of recursive Classification

If we linearly increase the amount of re-classified DAs, we can see from table 3 that the classification time also increases linearly. Additionally, the accuracy increases and reaches 58.78% by updating 40 segments. This seems to be a good trade-off between a maximized accuracy and a minimized classification time. Figure 2 visualizes the frequency of the updated history segments and we can see that - using 40 segments for the back-propagation - the labels only change within about 60 segments. This corresponds to a worst latency of **about 120 seconds**. After this processing time, the label of the DA reaches a stable state.

Despite the increased accuracy in the on-line system, we still have a degradation of 5.39% compared to the development system. This is most likely justified because of the wrongly (pre-)classified DA labels which confuses the classifier.

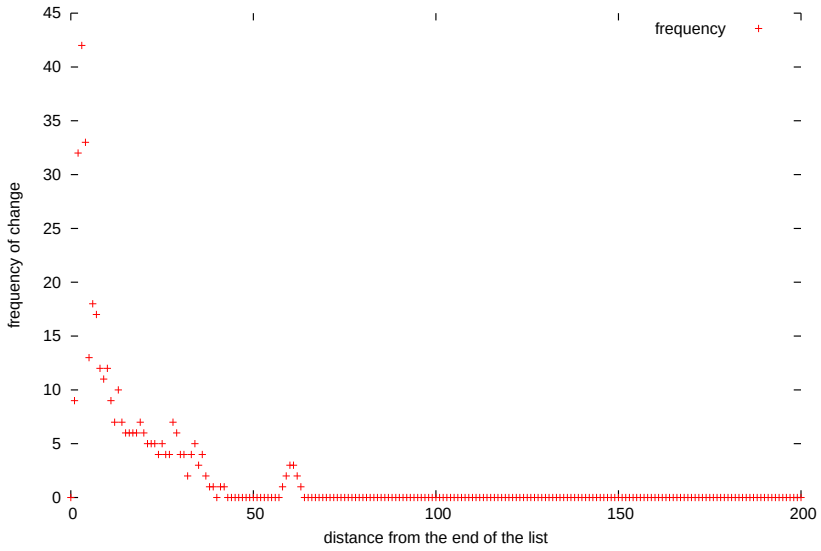

Fig. 2. Statistics of History Updating

6 Conclusions

We have described the design and results of a dialog act labeling approach that uses recursive classification embedded in an on-line system to increase its detection performance. We have seen that the system works and in fact improves its accuracy up to 58.78% which decreases the performance loss from the transportation to the on-line environment from 10% to 6%. Furthermore, we found out that the latency of such a system reaches from about 2 seconds up to 120 seconds where the first value corresponds to pre-classified labels with low accuracy and the latter corresponds to stable classified DAs with higher accuracy values. But, a latency of just 76 segments yields in an improvement of about 50% of the overall improvement.

6.1 Future Work

Our next planned steps are to integrate a prosodic feature set into the classification step which is by definition more robust towards on-line applications

than lexical and dynamic features. This should at least increase the baseline performance. Furthermore, we plan to use the dialog act segmentation system developed by [8] to implement a tool that is able to do both, dialog act segmentation and classification on its own.

7 Acknowledgment

This work is supported by the European IST Programme Project FP6-0033812 (AMIDA), Publication ID - AMIDA-26. This paper only reflects the authors views and funding agencies are not liable for any use that may be made of the information contained herein.

References

1. Lesch, S., Kleinbauer, T., Alexandersson, J.: "Towards a decent recognition rate for the automatic classification of a multidimensional dialogue act tagset.", In Workshop notes of the Fourth IJCAI Workshop on Knowledge and Reasoning in Pracical Dialogue Systems, 2005
2. Jeremy Ang, Yang Liu, Shriberg, E.: "Automatic Dialog Act Segmentation and Classification in Multiparty Meetings" Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP apos;05). IEEE International, Volume 1, Issue , March 18-23, 2005 Page(s): 1061 – 1064
3. Shriberg, E., Stolcke, A., Hakkani-Tür D., Tür, G.: "Prosody-based automatic segmentation of speech into sentences and topics", In, Speech Commun., Volume 32, 2000, Page(s): 127–154
4. Verbree, D., Rienks, R., Heylen, D.: "DIALOGUE-ACT TAGGING USING SMART FEATURE SELECTION; RESULTS ON MULTIPLE CORPORA", Spoken Language Technology Workshop, 2006. IEEE, Dec. 2006 Page(s): 70 – 73
5. Carletta, Jean, Ashby, S., Bourban, S., Flynn, M., et al: The AMI Meeting Corpus In: Proceedings of the Measuring Behavior 2005 symposium on "Annotating and measuring Meeting Behaviour", 2005
6. "AMIDA: Augmented Multiparty Interaction with Distance Access", Deliverable D5.2: Report on multimodal content abstraction. Technical report, Brno University of Technology, DFKI, ICSI, IDIAP, TNO, University of Edinburgh, University of Twente and University of Sheffield (2007)
7. Witten, I., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques 2nd volume, San Francisco: Morgan Kaufmann, 2005
8. Op den Akker, H., Schulz, C.: "Exploring Features and Classifiers for Dialogue Act Segmentation", Submitted