Automatic Summarization of Spoken Dialogues in Unrestricted Domains

Klaus Zechner

November 2001

CMU-LTI-01-168

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Thesis Committee:
Alex Waibel, Chair
Jaime Carbonell
Alon Lavie
Vibhu Mittal (Google)

This research was supported in part by the VerbMobil project of the Federal Republic of Germany, ATR Laboratories in Japan, and the US Department of Defense.

For My Wife
Michelle
With Love

Waasst wiast rutschst, wannst gehst?
Wiast segn, wiast rutschst, wannst foahrst!
Automobilist's advice, Vienna, Austria

Curiously deep, the slumber of crimson thoughts:
While breathless, in stodgy viridian,
Colorless green ideas sleep furiously.
J.Hollander

Acknowledgements

When I was about ten years old, a dear aunt of mine gave me the *Sprachbastelbuch* for my birthday. Decades later, this still is a quite popular and renowned book in German speaking countries. It contains a wealth of poems, stories, games, etc., whose sole purpose it is to *play with language*. I was fascinated and amazed by this book; maybe it was then, that I first discovered my true intellectual love, the human language. My history of education has been a long, if not convoluted story, starting college with computer science (my "second love", in a sense), adding linguistics a while later, and eventually winding up concentrating on their combination: computational linguistics. In that course, I earned some Master's degrees, until I embarked on this project, my Ph.D. thesis. To put it into a nutshell: It's been a long time, but a very enjoyable one, and more than that — a dream come true.

I owe a lot to the members of my thesis committee, starting with my main advisor Alex Waibel, who always encouraged and supported my interests in the various areas of research I have been engaged in throughout these years. He always challenged me with new scientific possibilities of my work, and although it consumed quite a bit of extra time, I am also grateful to him for allowing my thesis to have some practical impact, in that I ported my system to a different platform for it to be integrated into the multi-modal GUI, programmed by Michael Bett, the "Meeting Browser". With Alon Lavie I worked on a regular basis; he has been always available and very helpful indeed, pointing out the strong and weak points of my ongoing research in a very succinct and brilliant way. From Jaime Carbonell's and Vibhu Mittal's respective expertise in the field of summarization, as well as in a more general sense of doing research, I also profited very much.

I want to acknowledge the people from the Interactive Systems Labs, both at Carnegie Mellon and at Karlsruhe university (where I spent two summers), first and foremost Michael Bett, who implemented the Meeting Browser, and with whom I closely collaborated in the integration of my DIASUMM system in this multi-media and multi-modal tool. I also am grateful to many interesting discussions with Klaus Ries, Marsal Gavaldà, and Jade Goldstein in these past years. Klaus Ries and Marsal Gavaldà, as well as Benjamin Han, I have to thank for providing me with various software that I used for the thesis (the speech act tagger, various incarnations of the SOUP parser and adaptations of Brill's POS tagger). Hua Yu helped me generously with the setup and configuration of various versions of the JANUS recognition toolkit, and Chad Langley, my office mate, assisted me in figuring out the intricacies of C4.5 training.

Many thanks go also to the six topic and relevance annotators who had the not so easy task to go over more than 20 dialogues and mark them up in considerable detail. Thanks also to Vicky MacLaren, who did the disfluency and speech act annotations for the dialogue corpus.

I deeply thank my family, in particular my parents, who have always encouraged and supported my long path as a student of various subjects in all ways, and who had the faith and trust that I would know what is best for me. But most of all, I thank my wife from the depth of my heart and soul for all her support in these past several years of my thesis work, for simply being who she is — it is to her that I dedicate this OPUS SCIENTIARUM of mine.

Abstract

While the majority of summarization research so far has focused on written documents (mostly news articles or scientific papers), this thesis addresses for the first time the challenge of automatically summarizing spoken dialogues in a variety of genres and without any restriction on domain.

To achieve the goal of spoken dialogue summarization, we implement a system (DiaSumm) using a multi-stage architecture with trainable components which addresses the dialogue-specific issues of summarization and which involves (i) speech disfluency detection and removal, (ii) identification and insertion of sentence boundaries, (iii) identification and linking of question-answer regions, (iv) topical segmentation, and (v) information condensation (ranking of relevant pieces of information with the maximum marginal relevance technique (MMR)). We can also optionally reduce the summary content in an orthogonal dimension by rendering only a subset of the phrases within a relevant sentence (typically, noun phrases).

For system development and evaluation, we use a corpus of 23 dialogue excerpts from four different text genres, totalling 80 topical segments, about 47000 words, or about 4 hours of recorded speech: English CallHome (informal, colloquial style), Group Meetings (task oriented, rather informal, colloquial), and dialogue oriented television shows: NewsHour and CrossFire (more formal, potentially partially scripted). The corpus had been manually transcribed and was annotated for topical boundaries and relevant text spans by six human annotators. Further, it was annotated for speech disfluencies and questions and their corresponding answers. We devise a word-based evaluation criterion, relative summary accuracy, which reflects how well the summary captures passages that were placed in man-made summaries by the largest number of annotators.

The global evaluation, performed on human transcripts, shows that for the two more informal genres (CallHome and Group Meetings), DiaSumm significantly outperforms a baseline using TF*IDF term weighting with MMR ranking only, while tying with the MMR baseline for the two more formal genres. Furthermore, except for the NewsHour corpus, both the MMR baseline and our DiaSumm system are significantly better than a LEAD baseline (first N words of each segment). Finally, when using speech recognizer output, our system can make successful use of speech recognizer confidence scores to focus on sentences which are more likely to be correctly recognized; thereby, the word error rate in summaries can be reduced significantly while relative summary accuracy improves on average.

Contents

List of Tables

List of Figures

"He said he was against it."
C.Coolidge, on being asked what a clergyman preaching on sin had said. ("Presidential summary")

Chapter 1

Introduction

1.1 Introduction

While the field of summarizing written texts has been explored for many decades, gaining significantly increased attention in the last five to ten years, summarization of spoken language is a comparatively recent research area. As the amount of spoken audio databases is growing rapidly, however, we predict that the need for high quality summarization of information contained in this medium will rise substantially. Summarization of spoken dialogues, in particular, may aid the archiving, indexing, and retrieval of various records of oral communication, such as corporate meetings, sales interactions, or customer support.

The purpose of this thesis is to explore the issues of spoken dialogue summarization and to describe and evaluate an implementation addressing some of the core challenges intrinsic to the task: the DIASUMM system.

We consider the following dimensions to be relevant for our research; the combination of these dimensions distinguishes our work from other work in the field of summarization:

- spoken vs. written language
- multi-party dialogues vs. texts written by one author
- unrestricted vs. restricted domains
- diverse genres vs. a single genre

The main challenges this work has to address, in addition to the challenges of written text summarization, can be summarized as follows:

- coping with speech disfluencies
- identifying the units for extraction
- maintaining cross-speaker coherence
- coping with speech recognition errors
- identifying coherent topical regions

Intrinsic evaluations of text summaries typically use sentences as their basic units. For our data, however, sentence boundaries are typically not available in the first place. So we devise a word based evaluation metric based on an average relevance score from human relevance annotations.

1.2 Thesis Statement

In this thesis, we show that in order to create good summaries of spoken dialogues, methods specifically aimed at this text genre have to be applied in addition to standard state-of-the-art techniques aimed at written texts (such as maximum marginal relevance (MMR) (Carbonell and Goldstein, 1998)). Specifically, the issues of multiple speakers, lack of clause and topic boundaries, speech disfluencies, and speech recognizer errors are being addressed. We show that for human transcriptions of dialogues from informal genres (CALLHOME and GROUP MEETINGS), the spoken dialogue summaries generated by the DIASUMM system significantly outperform two baselines: a LEAD summarizer and a MMR text summarizer, using a word-based evaluation metric based on human relevance judgements.

1.3 Thesis Contributions

The main contributions of this thesis can be summarized as follows:

- addressing specific issues of spoken dialogues: lack of clause and topic boundaries; speech disfluencies; multiple speakers; speech recognizer errors

- evaluations of summaries for different genres of spoken dialogues
- creation of a summarization system (DIASUMM) for spoken dialogues which works both on speech recognizer hypotheses and on manually created transcripts and which is embedded in a graphical user interface for recording, transcribing, archiving, and summarizing of spontaneous multi-party conversations (Meeting Browser)
- multiple levels of content annotations (indicative keywords, noun phrase summaries, cleaned summaries): these can facilitate interactive drill-down summarization by providing customizable presentation of a dialogue's content
- the creation of a word-based evaluation metric which reflects the relevance annotations of the human coders and is easily usable for both ideal (human made) and speech recognizer transcripts
- a user study evaluating reading/answer-time and answer accuracy for different types of summaries
- a method for reducing word error rate in summaries from speech recognizer transcripts which also improves summary accuracy

1.4 Ethical Considerations

This may be an unusual section for a Ph.D. thesis but in my opinion, it should not. It is my firm belief that science is an endeavor of human beings — not only of scientists themselves, but of all humanity. The direction, the goals, the foci of scientific research have to be, therefore, a reflection of the desires of society in which it is conducted and the world at large.

In my opinion, there has to be a fruitful process of interaction between the research community and the (so-called) laymen in society who may not understand all the intricate details of a particular field of science but who are entitled to know and understand what is going on, at least in a general and principled sense. This process of information, interaction, and discussion on a collective level will hopefully allow, on the one hand, more critical self reflection on the methods, assumptions, and contents of scientific research, and also, on the other hand, a better acceptance, appreciation, and critical observation by those who are not directly involved in scientific research.

Ultimately, everyone pays for publicly funded scientific work (by means of taxes), and should therefore also have a genuine interest in the whats, hows, and whys of current scientific research. Corporate funding (and to some extent also: funding by military branches of the government) can be a slightly different matter, in that the interests reflected by the funding body may not necessarily represent the public interests, and/or may be kept confidential. Still, I believe that this does not spare scientists of the necessity of reflection upon their own work and of trying to put it in the context of the society they live in and putting in every effort in assessing the potential consequences of the results of their research.

This section should serve exactly this purpose: To create a basis for providing at least brief information about the current status of the field this research is a part of and for potential future developments and applications thereof.

1.4.1 Potential applications of this work

First, I see this work as a part of the large field of information retrieval and information extraction which has massively expanded in the past decade, due to the "information explosion" on the internet and other modern media. More specifically, summarization tries to fulfill the need of many users (people who seek information) to quickly glance over the most salient portions of any document — in the most general sense, such a document can be composed of text, but also of pictures, audio, and/or video information. Thus, this application can best be described with information compression or information condensation.

Second, and more specifically, spoken dialogue summarization aims at the automatic creation of (searchable) condensed records of all kinds of oral communication, such as interactions in the customer service field, but also in corporate meetings, or phone conferences. At the current state of technology, particularly when speech recognition is involved, only a rather low level of accuracy and reliability can be expected from automatic dialogue summarization systems. Some important information may be missed due to speech recognizer errors or just general shortcomings of the system. Also, one always has to keep in mind that one user's information needs usually do not exactly match another user's, and that therefore the notions of importance, relevance, and salience, are, at least to some extent, user-relative. However, realistically, systems could be built which are customized for focusing on particular topics, where their performance might be considerably better than a baseline general domain dialogue summarization system as we are presenting here in this work.

1.4.2 Privacy issues

If we consider dialogues that are beyond purposefully public conversations (such as interviews in radio or TV shows), we must face the issue of personal privacy protection. If corporate meetings were to be recorded, archived, summarized etc. on a regular basis, the question becomes relevant of who is allowed to access these records, and even the more basic question of whether everyone agrees to be recorded for all or some meetings. Furthermore, how long are these recordings kept in some archive? Will they be deleted and only kept in written, transcribed, summarized form?

Similar questions arise when dealing with secretly taped conversations, such as automatic surveillance by law enforcement or other federal agencies. Naturally, the question of consent will be moot in these cases, but the better the tools for automatically analyzing these conversations become, the more the public has to be aware of these powers and potential threats to privacy.

In my opinion, there is no intrinsically good or bad application of this technology *per se*, but it is important to keep its potentials and powers in the public eye and to allow for public discussion of when, where, and under what circumstances any particular summarization technology may be used and what restrictions should be imposed on its usage.

1.5 Thesis Outline

The organization of this thesis is as follows: Section 1.6 provides the motivation for our research, introducing and discussing the main challenges of spoken dialogue summarization, followed by a section on related work (section 1.7). Chapter 2 describes the corpus we use to develop and evaluate our system, along with the procedures employed for corpus annotation, and evaluations of inter-coder agreement. The system architecture and its components are described in detail in chapter 3, along with evaluations thereof. Chapter 4 presents the global comparative evaluation of our approach, an evaluation about the effect of topical segmentation on system performance, evaluations related to automatic word error rate reduction in dialogue summarization, evaluations concerning the local coherence gain by using cross-speaker information linking, and a user study comparing the informativeness of a gold standard, two baselines, and two versions of the DIASUMM system. Finally, we conclude the thesis with a discussion of our results and contributions and directions for future research in this field (chapter 5).

1.6 Motivation

Consider the following example from a phone conversation drawn from the English CALLHOME database (LDC, 1996). It is a transcript of a conversation between two native speakers of American English; one person is in the New York area (speaker A), the other one (speaker B) in Israel. It was recorded about a month after Yitzhak Rabin's assassination in the Fall of 1995. This dialogue segment is about one minute of real time; each turn is marked with its number and with the speaker label; noises are removed to increase readability. The turns are based on automatic segmentation of the audio stream by means of a silence detection algorithm.

```
 1 a: oh
 2 b: they didn't know he was going to get shot but it was at a
      peace rally so i mean it just worked out
 3 b: i mean it was a good place for the poor guy to die i mean
      because it was you know right after the rally and everything
      was on film and everything
 4 a: yeah
 5 b: oh the whole country we just finished the thirty days
      mourning for him now you know it's uh oh everybody's still
      in shock it's
 6 a: oh
 7 a: i know
 8 b: terrible what's going on over here
 9 b: and this guy that killed him they show him on t v smiling
      he's all happy he did it and everything he isn't even sorry
      or anything
10 a: there are i
```

```
11 b: him him he and his brother you know the two of them were
      in it together and there's a whole group now it's like a
      a conspiracy oh it's eh
12 a: mm
13 a: with the kahane chai
14 b: unbelievable
15 b: yeah yeah it's all those people yeah you probably see them
      running around new york don't you they're all
16 a: yeah
17 a: oh yeah they're here
18 b: new york based yeah
19 a: oh there's
20 a: all those fanatics
21 a: like the extreme
22 b: oh but
23 b: but wh- what's the reaction in america really i mean i mean
      do people care you know i mean you know do they
24 a: yeah mo- most pe- i mean uh
25 a: i don't know what commu- i mean like the jewish community
26 a: a lot e- all of us were
27 a: very upset and there were lots all the
28 b: yeah
29 a: like two days after did it happen like on a sunday
30 b: yeah it hap- it happened on it happened on a saturday night
```

We first show the output of a MMR (maximum marginal relevance) summarizer[1] (as used for written texts) for this segment (size=30% of original, 85 of 284 words):[2]

```
2 b :  They didn't know [...]
3 b :  I mean it was a good place for the poor guy to die I mean because
       it was you know right after the rally and everything was on film and
       everything
5 b :  Oh the whole country we just finished the thirty days mourning for
       him now you know it's uh oh everybody's still in shock it's
23 b : But what's the reaction in america really I mean I mean do people
       care you know I mean you know do they
```

[1]The details of this summarization method are described in section 3.6. MMR basically ranks sentences according to a relevance criterion while minimizing redundancy.

[2]When the length limit is reached during summary generation, the lowest ranking turn in the summary is cut off prematurely at the desired length; this is indicated by: [...].

By looking at this summary and the original transcript, we can readily identify some of the phenomena that are causing difficulties for conventional summarizers for written texts:

- some turns (e.g., turns 3, 5, 23) contain many disfluencies which (a) make them hard to read and (b) reduce the relevance of the information contained therein
- some (important) pieces of information are distributed over a sequence of turns (e.g., turns 5-8: "it's – terrible"); this is due to a silence based segmentation algorithm that causes breaks in logically connected clauses — a traditional summarizer might render these sequences incompletely, as seen in this example summary
- some turns are quite long (e.g., 3, 5, 9) and contain several sentences; a within-turn segmentation seems necessary, particularly if some of the sentences have a much higher relevance than others within the same turn
- some of the information is constructed interactively by both speakers; the prototypical cases are question-answer pairs (e.g., turns 23-24ff., turns 29-30) — a traditional text summarizer might miss either question or answer and hence produce a less meaningful and less coherent summary

We now present an improved summary of the above text, where (a) disfluencies are removed, (b) sentence boundaries match boundaries of extraction units (in some cases, several sentences are still extracted together, however), and (c) question-answer pairs are linked (marked with Q and A in this example).[3] This summary has the same length as the summary above (85 words), but is clearly more readable, coherent, and also represents the interactive nature of the dialogue much better:[4]

```
2 b :  It just worked out / it was a good place for the poor guy to die /
       because it was right after the rally / and everything was on film /
       the whole country we just finished the thirty days mourning
```

[3]For the sake of this illustration, (a) and (b) were performed manually, but (c) automatically by the DIASUMM system; we further increased the weight of detected questions and answers to increase the likelihood of them showing up in a summary.

[4]Note that the turn-IDs do not correspond to those in the example above.

```
        for him now / everybody's still in shock
   11 b : It's unbelievable / it's all those people you [...]
Q  15 b : What's the reaction in america / do people care
A  16 a : Yeah
A  17 a : The jewish community all of us were very upset
Q  18 a : Did it happen on a sunday
A  19 b : It happened on a saturday night
```

The goal of this thesis is to fully automatically create spoken dialogue summaries that have the properties of being readable, concise, and coherent. We believe that in addressing the main issues mentioned above, which are intrinsic to spoken dialogue summarization, we can indeed make progress towards this goal. We shall discuss these issues and ways to address them in chapter 3 of this thesis, but first we will provide a brief overview in the following subsections. We want to stress beforehand, though, that the originality of our work should not be seen in the particular implementation of the summarization system and its individual components, but rather in their selection and specific composition to address the issues at hand in an effective and also efficient way.

Our DIASUMM system will have a MMR component at its core, but will also have a set of additional modules which address these issues of spoken language dialogue. In the global system evaluations, we will compare this core MMR summarizer, as well as a LEAD based summarizer, against the complete DIASUMM system.

1.6.1 Disfluency detection

The two main negative effects speech disfluencies have on summarization are that they (i) decrease the readability of the summary and (ii) increase its non-content noise.

In particular for informal conversations, the percentage of disfluent words is quite high, typically around 15% of the total words spoken (see section 2.1). This means that this issue should, in our opinion, be addressed to improve the quality (readability and conciseness) of the generated summaries.

Let us look at an example here to show the potential effect of a disfluency detection component (turn 1: before, turn 1′: after disfluency detection and removal):

```
 1: so did he i mean did did they invite um you know the mortons
```

```
1': did they invite the mortons
```

In section 3.3 we shall present a multi-stage approach for identifying the major classes of speech disfluencies in the input of the summarization system, such as filled pauses, repetitions, and false starts. All detected disfluencies are marked in this process and can be selectively excluded during summary generation.

1.6.2 Sentence boundary detection

Unlike written texts, where punctuation markers clearly indicate clause and sentence boundaries, spoken language is generated as a sequence of streams of words, where pauses (silences between words) do not always match linguistically meaningful segments: a speaker can pause in the middle of a sentence or even a phrase, or, on the other hand, might not pause at all after the end of a sentence or clause.

This mismatch between acoustic and linguistic segmentation is reflected in the output of a speech recognizer which typically generates a sequence of speaker turns whose boundaries are marked by periods of silence (or non-speech). As a result, one speaker's turn may contain multiple sentences, or, on the other hand, a speaker's sentence might span more than one turn. In a test corpus of 5 English CALLHOME dialogues with an average length of 320 turns, we found on average about 30 such continuations of logical clauses over automatically determined acoustic segments per dialogue.

We provide a short example here, where sentence boundaries do not match turn boundaries in turns A1 and A2 of the same speaker A; ideally, the sentence boundary detection module's output should look like turns A1', A2':

```
 A1: are they coming did you invite
 A2: the mortons for the wedding

A1': are they coming
A2': did you invite the mortons for the wedding
```

The main problem for a summarizer would thus be (i) the lack of coherence and readability of the output because of incomplete sentences and (ii) extraneous information due to extracted units consisting of more than one sentence.

In section 3.3.7 we describe a component for sentence segmentation which addresses both of these problems.

1.6.3 Distributed information

Since we have multi-party conversations as opposed to monologues, sometimes the crucial information is found in a sequence of turns from several speakers — the prototypical case being a question-answer pair, as shown in the following examples:

```
A1: when are the mortons arriving
B2: they will be here saturday morning

A3: have you called your brother yet
B4: no i haven't
```

If the summarizer were to extract only the question or only the answer, the lack of the corresponding answer or question would often cause a severe reduction of coherence in the summary.

In some cases, either the question or the answer is very short (e.g., B4) and does not contain any words with high relevance which would yield a substantial weight in the summarizer. In order not to lose these short sentences at a later stage, when only the most relevant sentences are extracted, we need to identify matching question-answer pairs ahead of time, so that the summarizer can output the matching sentences during summary generation. We describe our approach to cross-speaker information linking in section 3.4 and evaluations concerning its effects on informativeness and coherence of summaries in section 4.4.

1.6.4 Topic segmentation

In many cases, spoken dialogues are multi-topical. For the English CALLHOME corpus, we determined an average topic length of about 1–3 minutes speaking time (or about 200–600 words). Summarization can be done faster and more concisely if it operates on smaller topical segments rather than on long pieces of input consisting of diverse topics. A further advantage of topic segmentation relates to the fact that in an interactive setting, topical keywords can be provided as a first approximation for a dialogue's content and can thus facilitate drill-down summarization.

(We realize this in the GUI of the Meeting Browser.)

While we have implemented a topic segmentation component as part of our system for these reasons, all of the major evaluations are based on the topical segments determined by human annotators for reasons of consistency and comparability. In section 4.2, though, we evaluate the effect of automatic and ideal topic segmentation on summary accuracy.

1.6.5 Speech recognition errors

Throughout most parts of this thesis, our simplifying assumption is that our input comes from a perfect speech recognizer, that is, we use human textual transcripts of the dialogues in our corpus. While there are cases where this assumption is justifiable, such as transcripts provided by news services in parallel to the recorded audio data, we believe that, in general, a spoken dialogue summarizer has to be able to accept corrupted input from an automatic speech recognizer (ASR), as well.

Our system is indeed able to work with ASR output; it is integrated in a larger system (*Meeting Browser*) which creates, summarizes, and archives meeting records and is connected to the JANUS speech recognition engine (Bett et al., 2000; Waibel et al., 2001). Further, we show in section 4.3 that we can use ASR confidence scores to (i) reduce the word error rate within the summary and (ii) increase the summary accuracy.

1.6.6 Other issues

We see the work reported in this thesis as the first implementation with in depth analysis and evaluation in the area of open domain spoken dialogue summarization. Given the large scope of this undertaking, we had to restrict ourselves to those issues discussed above which are, in our opinion, the most important for the task at hand. A number of other important issues for summarization in general and for speech summarization in particular are either simplified or not addressed in this thesis and left for future work in this field. In the following, we briefly mention some of these issues, indicating their potential relevance and promise.

Anaphora resolution

While it is certainly desirable, for the sake of increased coherence and readability, to employ a well-working anaphora resolution component, this issue is not specific to the task at hand; we did not implement a component for anaphora resolution in the context of this thesis. However, we do believe in its potential to enhance the quality of the summaries generated, particularly for genres such as CALLHOME with a more pronounced pronominal style.

Discourse structure

Previous work indicates that information about discourse structure from written texts can help identifying the more salient and relevant sentences or clauses for summary generation (Marcu, 1999; Miike et al., 1994). (That the reverse might be true as well, was shown, e.g., by Stifelman (1995).) However, much less exploration has been done in the area of automatic analysis of discourse structure for non task-oriented spoken dialogues in unrestricted domains, such as CALLHOME (LDC, 1996). Research for those kinds of corpora reported in (Stolcke et al., 2000; Levin et al., 1999; Ries et al., 2000) focuses more on detecting localized phenomena such as speech acts, dialogue games, or functional activities. We conjecture that there are two reasons for this: (i) free flowing spontaneous conversations have much less structure than task-oriented dialogues; (ii) the automatic detection of hierarchical structure would be much harder than it is for written texts or dialogues based on a pre-meditated plan.

While we believe that in the long run attempts to automatically identify the discourse structure of spoken dialogues may benefit summarization much like for the case of documents written by a single author, in this thesis, we greatly simplify this matter and exclusively look at local contexts of cross-speaker coherence where speakers interactively construct shared information (the question-answer pairs).

Prosodic information

A further simplifying assumption of this work is that prosodic information is not available, with the exception of start and end times of speaker turns.

It seems likely that using additional prosodic information, such as stress, pitch, and intra-turn pauses, could improve both the different system components individually, as well as the overall summarization system.

1.7 Related Work

In this section we discuss related work, as far it is pertaining to the topic of summarization in general, and spoken language summarization in particular. Related work concerning the dialogue-specific components of DIASUMM is discussed in the respective sections of chapter 3.

1.7.1 Traditional text summarization

While early research in automatic summarization dates as far back as to the late 1950's (Luhn, 1958), there has been a renewed and intensified interest in this area in the past decade or so, mainly due to the exponential growth of on-line data through newswire services and the emergence of the World Wide Web.

There are several dimensions which have to be considered when talking about summarization (cf. e.g. (Hovy and Marcu, 1998; Sparck-Jones and Endres-Niggemeyer, 1995; Mani and Maybury, 1999)), such as the following (we emphasize the primary foci of our DIASUMM system in bold):

- **extracts** vs. abstracts: While extracts are created by pure extraction of pieces of the original text (mostly: sentences or clauses, sometimes keywords and/or keyphrases), abstracts are *generated* from some sort of semantic representation which reflects the logical structure of the text: the former can be done with entirely statistical methods (possibly enhanced with some linguistic knowledge), the latter requires not only a "deep" understanding of the text — which is currently infeasible unless one works in a very limited domain only (Reimer and Hahn, 1997) — but also a generation component which produces intelligible text from the formal representation (Radev and McKeown, 1998).
- indicative vs. **informative**: Indicative summaries are meant to give the user a rough idea about the main points of a text; these are typically used for tasks such as text classification or information retrieval; informative summaries

should represent the most relevant information in a text and be able to serve as "surrogates" for the complete text.

- **generic** vs. query-driven: In the generic case, the summary should provide an unbiased view of the most relevant information in a text, if it is a query-driven summary, it should reflect the specific interests of this user by focusing on the query.

- **single** vs. multiple documents: Is there one text or several sources to summarize simultaneously? Multi-document summarization usually requires a much higher compression rate, along with a need for elimination of redundant information (Goldstein et al., 2000; Radev, Jing, and Budzikowska, 2000).

- background vs. just-the-news: In some instances, summarizers might have to be able to distinguish between these two kinds of information (specifically relevant for newswire data), e.g., to alert users to events which have not been reported in previous updates.

- single vs. **multiple topics**: Most short newswire articles (and research papers) will be mono-topical; however, there are many texts where this simplifying assumption does not hold and for which methods have to be established to reflect the multi-topicality in the summary.

- single vs. **multiple speakers**: The majority of text documents summarized will have a single speaker or writer; however, there are also interviews, discussions, conversations etc. where the information is distributed among multiple participants and sometimes is constructed by their interaction (e.g., by a question-answer pair).

- text-only vs. **multi-modal**: Summarization research so far almost exclusively focused on the written domain; in recent years, several research groups have started to explore how to summarize multi-modal and multi-media input (Waibel, Bett, and Finke, 1998; Waibel et al., 2001; Hirschberg et al., 1999; Valenza et al., 1999).

- **selecting sentences/clauses** vs. condensing within sentences: There has been a recent surge of research on trainable systems which can reduce the information *within* a sentence or a clause, whereas the mainstream of summarization research clearly has been concerned with sentence (or clause, paragraph) selection only. While (Jing, 2000) uses information from syntactic parses, context, and corpus statistics, (Knight and Marcu, 2000) use a noisy-channel and a decision tree model based on aligned parse trees of parallel corpora of (Text, Abstract) pairs. A similarly inspired approach was taken by (Banko, Mittal, and Witbrock, 2000; Berger and Mittal, 2000) who generate headlines automatically from news stories or Web pages, using a paradigm based on statistical machine translation. The work by (Hori and Furui, 2000) focuses on broadcast news caption reduction; they use a combination of salience features and a language model to achieve the goal of sentence compression.

In terms of approaches for summarization, three main directions have been pursued so far: (i) knowledge-intensive summarization (e.g., (Reimer and Hahn, 1997)) which is aiming at accurate text condensation at the disadvantage of working in a (very) limited domain only; (ii) discourse-based summarization (e.g., (Marcu, 1997)) where the most relevant pieces of a text are determined by means of a (potentially shallow) analysis of the discourse structure; and (iii) statistical summarization (e.g., (Kupiec, Pedersen, and Chen, 1995; Aone, Okurowski, and Gorlinsky, 1997)) where easily derivable and shallow features such as word frequency, cue words/phrases etc. are used to determine the most relevant passages in a text: this method does not need any (or only very little) knowledge engineering and is usually domain independent, at the expense of being able to create only a collection of text extracts which may yield a lower readability or acceptance by users.

In recent years there have been several efforts at combining these different approaches, mostly by adding some linguistic knowledge to a statistical summarization "backbone", e.g., using (Miller et al., 1993)'s WordNet to infer conceptual information (Barzilay and Elhadad, 1997), or noun phrase parsers (Barzilay and Elhadad, 1997) to identify heads of compound nouns.

It also should be mentioned that the field of information extraction (IE) — which had been given a significant push by TIPSTER's Message Understanding Conferences (MUCs) (Altomari and Currier, 1996; Gee, 1996; Sundheim, 1996) — is somewhat related to the task of summarization. The emphasis is different, however, in that IE is focusing on identifying *pre-specified* kinds of information in texts of a usually quite narrow domain (e.g. company mergers). The result of an IE task is usually a database of slot-filler templates which could be used to generate a summary about a given document (or a set of documents). According to the summarization dichotomies stated above, IE is looking for a query-driven, informative, just-the-news, (usually single text), mono-topical summary (in template-format).

Since in this thesis, we summarize texts in unrestricted domains, we use a statistical method for summary creation (the “core” component of the DIASUMM system), which uses term frequencies, inverse document frequencies, and Maximum Marginal Relevance (Carbonell and Goldstein, 1998), the latter being a method which performed very well at the first (and until 2001 only) objective multi-system summarization evaluation of TIPSTER (Mani et al., 1998). This component will serve as the major baseline system for this thesis.

Other shallow methods and features used in previous work include the following:

- Location: Many applications use information about the relative location of sentences to score their relevance for a summary (Kupiec, Pedersen, and Chen, 1995; Teufel and Moens, 1997; Hovy and Lin, 1997). It is noted, e.g., that beginnings of paragraphs tend to contain more relevant information than their middle sections. Some researchers also found that just using the “lead” (i.e., the beginning of a document) produces not only more acceptable or readable, but sometimes also more informative summaries (Brandow, Mitze, and Rau, 1995; Wasson, 1998). In this work, the LEAD summaries will serve as a second baseline for the global evaluations. Here, the LEAD starts from the beginning of a topical segment within a dialogue.

- Title: Titles can be conceived of as “miniature summaries” and hence are frequently used by summarization systems (Kupiec, Pedersen, and Chen, 1995; Hovy and Lin, 1997). Either they are included in the summary itself and/or they are used as “query” to increase the weights of words in the documents which also appear in the title.

- Cue words/phrases: In written texts, particularly in scientific papers, there are many cues that indicate sentences which either should or should not be included in summaries (trigger or stigma words). Usually they are determined manually (Teufel and Moens, 1997) by inspecting the texts and possibly a human generated gold standard summary based on extracts of sentences from the original texts.

- Text cohesion: (Salton et al., 1994; Hovy and Lin, 1997; Boguraev and Kennedy, 1997; Barzilay and Elhadad, 1997) build *lexical chains* between either lexically or semantically related words in a document. The hypothesis is that a summary should include those sentences that are part of *strong chains*. To identify members of the chains, cooccurrence, proximity, anaphoric reference, or concept-relatedness (e.g. as determined by the WordNet hierarchy (Miller et al., 1990)) are used.

To address the issue of multi-topicality, several authors (Boguraev and Kennedy, 1997; Barzilay and Elhadad, 1997) use (Hearst, 1997)'s TextTiling approach; this is also the approach taken in my DIASUMM system (cf. section 3.5).

A critical and largely unresolved issue in summarization is the question of how to best evaluate a summarization system. There are two main perspectives on this issue:

1. Intrinsic evaluation: one is interested in determining the "quality" of a summary in itself; this is usually done by having human subjects evaluate the summary. One can also compare it to a (manually created) "gold summary" (or: gold standard summary). If the summary just contains sentences from the text, precision and recall can be computed easily. A measure such as the 11-pt-avg precision (Salton and McGill, 1983) can further be used to account for the precision/recall-tradeoff of different summary lengths. So far, most studies were done in this paradigm (Edmundson, 1969; Kupiec, Pedersen, and Chen, 1995; Marcu, 1997). Another possibility to evaluate the intrinsic quality of a summary is to ask a user a set of questions which should cover the most essential pieces of information in a given text and compare the answer accuracy against some baseline(s). The TIPSTER SUMMAC evaluation had one part which was conducted along these lines (Mani et al., 1998). More recently, there have been proposals to extend the dimensions of summary evaluations from mere informativeness to other criteria such as grammaticality, cohesion, or organization (NIST, 2001). While these dimensions might be harder to evaluate automatically than just mere presence of some key concepts from the original text, they might help to ensure that a summary with high ratings in all dimensions is more likely to be useful to a human user.

2. Extrinsic evaluation: here, the question is how good one can perform a particular task when using a summary as a substitute for a full document. This

was the focus of the TIPSTER SUMMAC evaluations (Mani et al., 1998) where summaries were used to (a) judge document relevance (query-specific), and to (b) classify a document to a certain topic (generic).

In this thesis, we focus on *intrinsic summary evaluation*: The automatically generated summaries will be compared against human-created summaries; to alleviate the problem of differences in relevance judgements (Rath, Resnick, and Savage, 1961), multiple coders and average relevance scores will be used.

1.7.2 Towards summarization of spoken language

There are two main areas which are exceptions to the focus on text summarization in past work: (i) summarization of task oriented dialogues in restricted domains, and (ii) summarization of spoken news in unrestricted domains. We shall discuss both of these areas in the following subsections, followed by a discussion of prosody-based emphasis detection in spoken language, and finally by research most closely related to the topic of this thesis.

Summarization of dialogues in restricted domains

During the past decade, there has been significant progress in the area of closed domain spoken dialogue translation and understanding, even with automatic speech recognition input. Two examples of systems being developed in that time frame are JANUS (Lavie et al., 1997) and VERBMOBIL (Wahlster, 1993).

In that context, several spoken dialogue summarization systems were developed, whose goal it was to capture the essence of the task based dialogues at hand. The MIMI System (Kameyama and Arima, 1994; Kameyama, Kawai, and Arima, 1996) dealt with the travel reservation domain and used a cascade of finite state pattern recognizers to find the desired information.

Within VERBMOBIL, a more knowledge-rich approach was used (Alexandersson and Poller, 1998; Reithinger et al., 2000). The domain here is travel planning and negotiation of a trip. In addition to finite state transducers for content extraction and statistical dialogue act recognition, they also use a dialogue processor and a summary generator which have access to a world knowledge database, a domain model, and a semantic database. The abstract representations built by this summarizer allow for summary generation in multiple languages.

Summarization of spoken news

Within the context of the TREC spoken document retrieval (SDR) conferences (Garofolo et al., 1997; Garofolo et al., 1999) as well as the recent DARPA Broadcast News workshops, a number of research groups have been developing multi-media browsing tools for text, audio, and video data, which should facilitate the access to news data, combining different modalities.

(Hirschberg et al., 1999; Whittaker et al., 1999) present a system that supports local navigation for browsing and information extraction from acoustic databases, using speech recognizer transcripts in tandem with the original audio recording. While their interface helped users in the tasks of relevance ranking and fact-finding, it was less helpful in the creating of summaries, partly due to imperfect speech recognition.

Valenza et al. (1999) present an audio summarization system which combines acoustic confidence scores with relevance scores to obtain more accurate and reliable summaries. An evaluation showed that human judges prefer summaries with a compression rate of about 15% (30 words per minute at a speaking rate of about 200 words per minute), and that the summary word error rate was significantly smaller than the word error rate for the full transcript.

Hori and Furui (2000) use salience features in combination with a language model to reduce Japanese broadcast news captions by about 30-40% while keeping the meaning of about 72% of all sentences in the test set. Another speech related reduction approach was presented recently by Koumpis and Renals (2000) (voicemail summarization into the Small Message format).

Prosody-based emphasis detection in spoken audio

While most approaches to summarizing of acoustic data rely on the word information (provided by a human or ASR transcript), there have been attempts to generate summaries based on emphasized regions in a discourse, using only prosodic features. Chen and Withgott (1992) train a Hidden Markov Model on transcriptions of spontaneous speech, labeled for different degrees of emphasis by a panel of listeners. Their "audio summaries" on an unseen (but rather small) test set receive a remarkably good agreement with human annotators ($\kappa > 0.5$). Stifelman (1995) uses a pitch based emphasis detection algorithm developed by Arons (1994) to find emphasized passages in a 13 minute discourse. In her analysis, she finds good agreement between these emphasized regions and the beginnings of manually marked discourse segments (in the framework of Grosz and Sidner (1986)). Although these are promising results, being suggestive of the role of prosody for determining emphasis, relevance, or salience in spoken discourse, we restrict the use of prosody to the turn length and inter-turn pause features in this thesis. We conjecture, however, that the integration of prosodic and word level information would be a fruitful research area that would have to be explored in future work.

Spoken dialogue summarization in unrestricted domains

Waibel, Bett, and Finke (1998) report results of their summarizer on automatically transcribed SWITCHBOARD data (Godfrey, Holliman, and McDaniel, 1992), the word error rate being about 30%. Their implementation used an algorithm inspired by MMR, but they did not address any dialogue or speech related issues in their summarizer. In a question-answer test with summaries of five dialogues, subjects could identify most of the key concepts using a summary size of only five turns. These results varied widely across five different dialogues tested in this experiment (between 20% and 90% accuracy).

Our own previous work (Zechner and Waibel, 2000a) addressed this combination of challenges of dialogue summarization with summarization of spoken language in unrestricted domains for the first time. We presented a first prototype of DIASUMM which addressed the issues of disfluency detection and removal, sentence boundary detection, as well as cross-speaker information linking.

This thesis extends and expands these initial attempts substantially, in that we are now focussing on (i) a systematic training of the major components of the DIASUMM system — enabled by the recent availability of a large corpus of disfluency annotated conversations (LDC, 1999b) — and (ii) the exploration of three more genres of spoken dialogues in addition to the English CALLHOME corpus (NEWSHOUR, CROSSFIRE, GROUP MEETINGS). Further, the relevance annotations are now done by a set of six human annotators, which makes the global system evaluation more meaningful, considering the typical divergence among different annotators' relevance judgments.

If I talk about language (word, sentence, etc.)
I have to use every day language.
L.Wittgenstein

Chapter 2

Corpus

We begin this chapter by describing and characterizing the corpus of dialogues from four different genres which we use to develop and evaluate the DIASUMM system. The second section describes the different dimensions for the corpus annotation, looks at the general differences in annotation style and one dialogue in more detail, and finally evaluates the inter-coder agreement on various annotation tasks.

2.1 Corpus Characteristics

Table 2.1 provides the statistics on the corpus used for the development and evaluation of our system. We use data from four different genres, two being more informal, two more formal:

- English CALLHOME and CALLFRIEND: from the Linguistic Data Consortium (LDC) collections (LDC, 1996), 8 dialogues for the `devtest`-set (8E-CH) and 4 dialogues for the `eval`-set (4E-CH).[1] These are recordings of phone conversations between two family members or friends, typically about 30 minutes in length; the excerpts we used were matched with the transcripts which typically represent 5–10 minutes of speaking time.

[1] We used the `devtest`-set corpus for system development and tuning, and set aside the `eval`-set for the final global system evaluation. For the other three genres, two dialogue excerpts each were used for the `devtest`-set, the remainder for the `eval`-set.

Table 2.1
Data characteristics for the corpus (average over dialogues).

Data set	8E-CH	4E-CH	NHOUR	XFIRE	G-MTG
formal/informal	informal	informal	formal	formal	informal
topics pre-determined	no	no	yes	yes	yes
dialogue fragments (total)	8	4	3	4	4
topical segments (total)	28	23	8	14	7
different speakers	2.1	2	2	6	7.5
turns	242	276	25	96	140
sentences	280	366	101	281	304
sentences per turn	1.2	1.3	4.1	2.9	2.2
questions (in %)	3.7	6.4	6.3	9.8	4.0
false starts (in %)	12.1	11.0	2.0	7.2	13.9
non-disfl. words in false starts (in %)	5.7	4.8	1.1	3.0	6.2
words	1685	1905	1224	3165	2355
words per sentence	6.0	5.2	12.1	11.3	7.7
disfluent (in %)	16.0	16.3	5.1	4.2	13.2
disfluencies	222	259	48	95	266
disfluencies per sentence	0.79	0.71	0.48	0.34	0.87
empty coord. conjunctions (in %)	30.3	30.4	64.8	50.7	24.3
lexicalized filled pauses (in %)	18.8	21.0	17.2	23.5	13.9
editing terms (in %)	3.6	1.6	3.4	5.7	3.3
non-lex. filled pauses (in %)	20.8	29.9	0.7	2.3	29.5
repairs (in %)	26.6	17.1	13.8	17.8	29.0

- NEWSHOUR (NHOUR): Excerpts from PBS's NEWSHOUR TV show with Jim Lehrer (recorded in 1998).
- CROSSFIRE (XFIRE): Excerpts from CNN's CROSSFIRE TV show with Bill Press and Robert Novak (recorded in 1998).
- GROUP MEETINGS (G-MTG): Excerpts from recordings of project group meetings in the Interactive Systems Labs at Carnegie Mellon University.

Furthermore, we used the Penn Treebank distribution of the SWITCHBOARD corpus, annotated with disfluencies, to train the major components of the system (LDC, 1999b).

From Table 2.1 we can see that the two more formal corpora, NEWSHOUR and CROSSFIRE, have longer sentences, more sentences per turn, and fewer disfluencies (particularly non-lexicalized filled pauses and false starts) than English CALLHOME and the GROUP MEETINGS. This means that their flavor is more like that of written text, and not so close to conversational speech typically found in the SWITCHBOARD or CALLHOME corpora.

Just by computing the relative part-of-speech (POS) tag frequency of the UH-tag (non-lexicalized filled pauses) we can discriminate well between formal and informal dialogues, both using human as well as automatic (ASR) transcriptions: for formal dialogues, the relative frequency of UH is below 1%, for informal dialogues, above 3% (see Figure 2.1).

We also tried to characterize the individual dialogues on the following two dimensions: (i) nominal style: this represents the ratio of nominal part-of-speech tags (NN, NNS, NNP, NNPS) to verbal POS tags (BES, HVS, VB, VBP, VBN, VBD, VBZ, VBG, MD); and (ii) pronominal style: the ratio of pronominal tags (PRP, PRP$) to nominal tags (NN, NNS, NNP, NNPS).[2] Figure 2.2 shows that the CallHome dialogues are less nominal (more verbal) and more pronominal in style than the dialogues from NewsHour and CrossFire (there are two exceptions, however); dialogues from the Group Meetings fall between these two classes. Again, these observations hold for both human as well as ASR transcriptions.

[2] See (Santorini, 1990; LDC, 1999a) for a description of the tags used for the Penn Treebank project and Appendix A for a listing of POS used in our system.

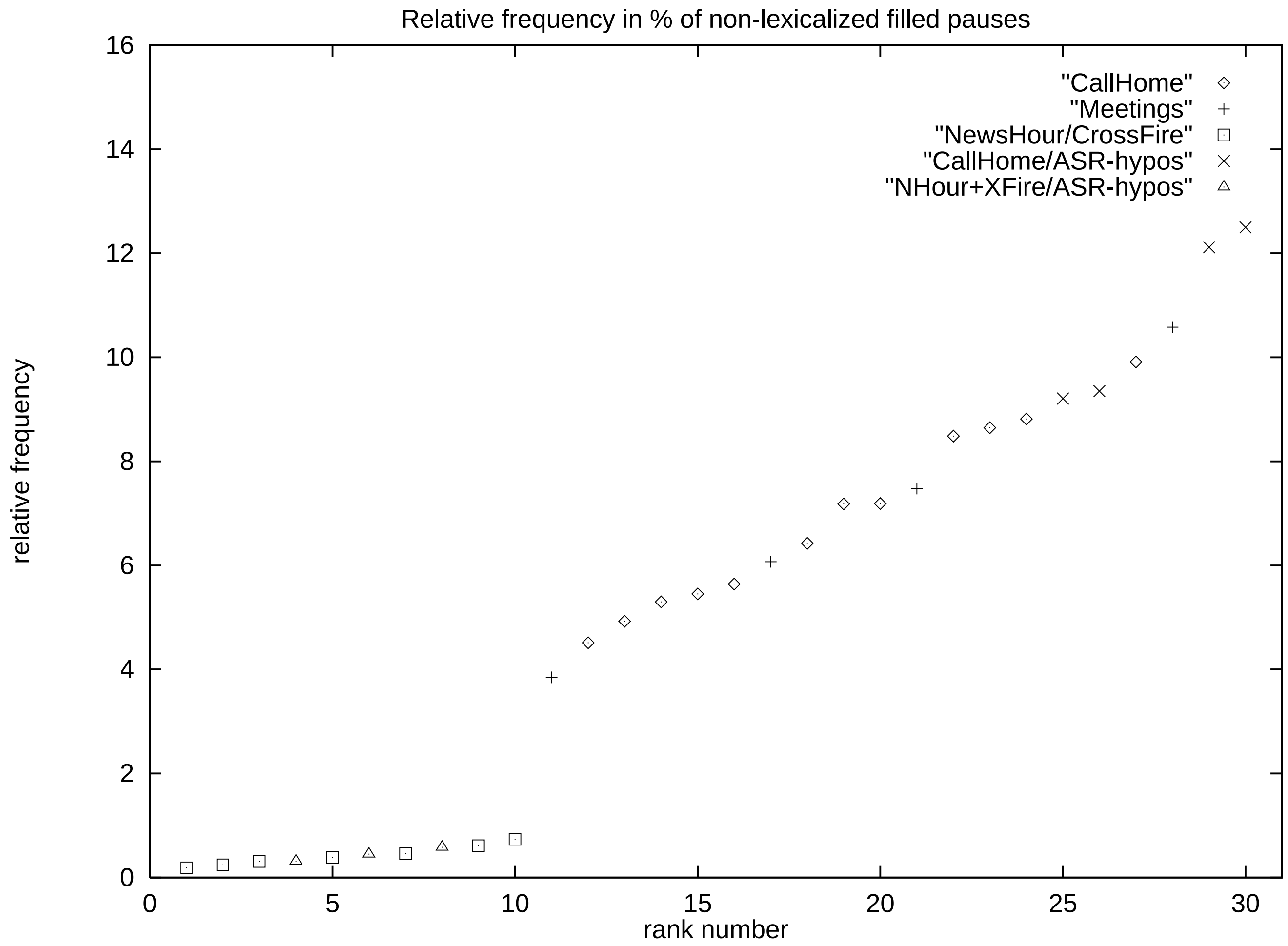

Figure 2.1
Relative frequencies of non-lexicalized fillers in different dialogues (both human transcripts and ASR transcripts).

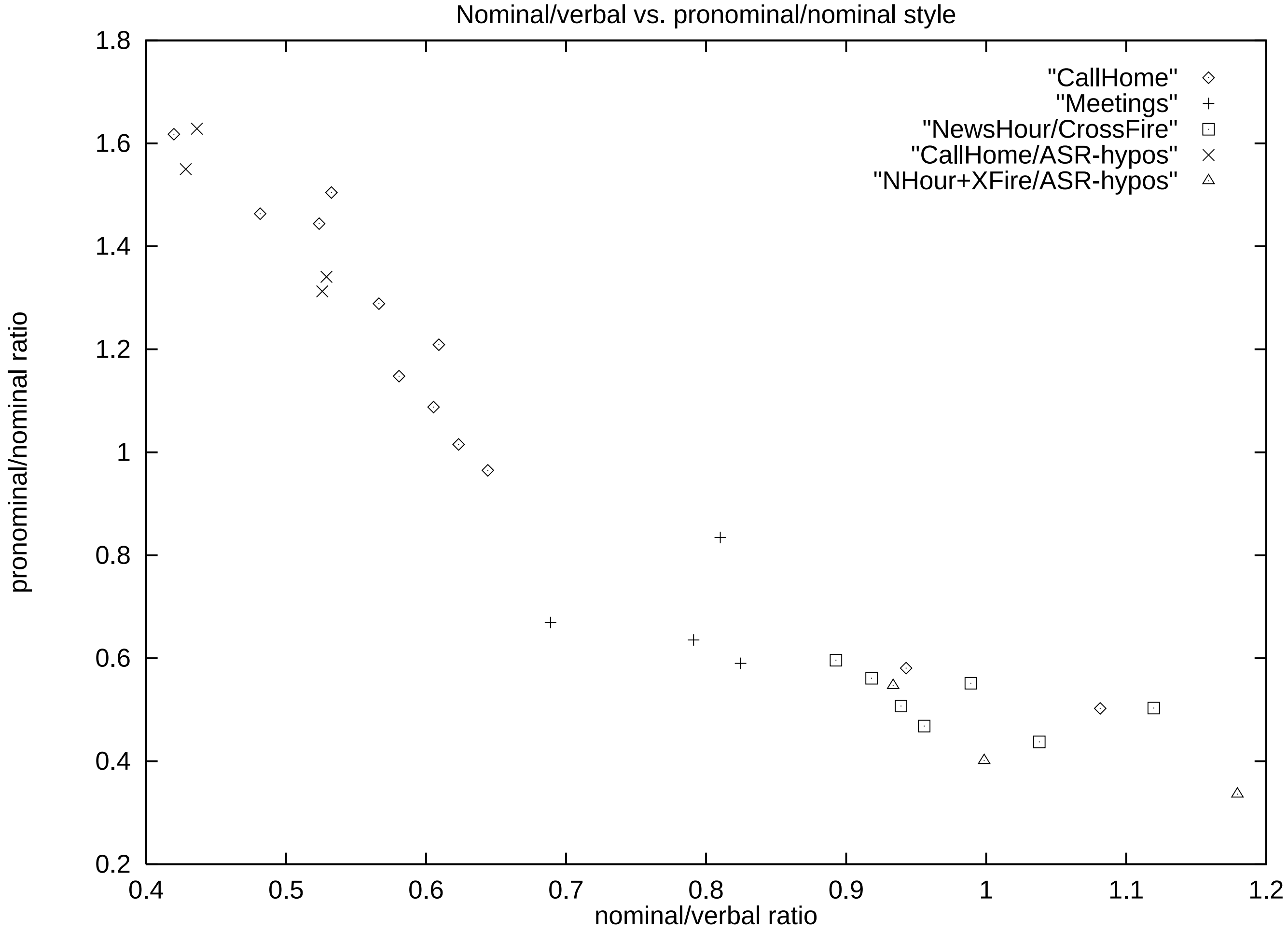

Figure 2.2
Contrastive comparison of different corpora (both human transcripts and ASR transcripts): nominal style vs. pronominal style.

2.2 Corpus Annotation

2.2.1 First annotation phase

All the annotations are performed on human generated transcripts of the dialogues. The CALLHOME and GROUP MEETING dialogues were automatically partitioned into speaker turns (by means of a silence heuristic), the other corpora were segmented manually (based on the contents and flow of the conversation). [3]

There were six human annotators performing the task; however, only four completed the entire set of dialogues. Thus, the number of annotations available for each dialogue varies from four to six.

Prior to the relevance annotations, the annotators had to mark topical boundaries, because we want to be able to define and then create summaries for each topical segment separately (as opposed to a whole conversation consisting of multiple topics).

For each topical segment, each annotator had to identify the most relevant information units (IUs), called *nucleus IUs*, and somewhat relevant IUs, called *satellite IUs*. IUs are often equivalent to sentences, but can span longer or shorter contiguous segments of text, dependent on the annotator's choice. The overall goal of this relevance mark-up was to create a concise and readable summary, containing the main information present in the topical segment. Annotators were also asked to mark the most salient words within their marked IUs with a '+', which would render a summary with a somewhat more telegraphic style ('+'-marked words).

We also asked that the human annotators stay within a pre-set target length for their summaries: the '+'-ked words in all IUs within a topical segment should be about 10–20% of all the words in this segment. The guideline is enforced by a checker program which was run during and after annotation of a transcript and which also ensured that no mark-up errors and no accidental word deletions occurred. Furthermore, this script also displays the currently marked ideal summary as a sanity check. Appendix B.1 provides an example from an annotated file.

[3] This fact may partically account for NewsHour and CrossFire turns being longer than CallHome and Group Meeting turns.

2.2.2 Creation of gold standard summaries

After the first annotation phase, where each coder worked independently according to the guidelines described above, we devised a second phase, in which two coders (from the initial group) were asked to create a common ground annotation, based on the majority opinion of the whole group. To construct such a majority opinion guideline automatically, we assigned weights to all words in nucleus-IUs and satellite-IUs and added all weights for all marked words of all coders for every turn.[4] The total turn weights were then sorted by decreasing value to provide a guide for the two coders in the second phase, on which turns they should focus their annotations for the common ground or gold standard summaries.

Other than this guideline, the requirements were almost exactly identical to Phase 1, except that (a) the pair of annotators was required to work together on this task to be able to reach a consensus opinion, and (b) the pre-set relative word length of the gold summary (10-20%) only applied to the nucleus IUs.

As for the topical boundaries, which obviously vary between different coders, a list of boundary positions chosen by the majority (at least half) of the coders in the first phase was provided.[5]

2.2.3 Annotation analysis

Nucleus and satellite statistics

Table 2.2 provides the statistics on the frequencies of annotated nucleus- and satellite-IUs. We make the following observations:

- on average, about 23% of all tokens were assigned to a nucleus-IU, and 5% to a satellite-IU; counting only the '+'-marked tokens, this reduces to about 11% resp. 2% of all tokens
- the average total lengths of nuclei and satellites vary widely across corpora:

[4] The weights were set as follows: n-IUs: 3.0 if '+'-marked, 2.0 otherwise; s-IUs: 1.0 if '+'-marked, 0.5 otherwise.

[5] In this gold standard phase, the two coders mostly stayed with these suggestions and changed less than 15% of the suggested topic boundaries, the majority of which being minor (less than two turns difference).

data set/ annotator	avg.nucl. length	avg.sat. length	nuc-tokens (in %)	nuc-marked (in %)	sat-tokens (in %)	sat-marked (in %)
LB	12.993	13.732	11.646	8.558	5.363	3.818
BR	16.507	14.551	11.978	8.339	10.558	7.645
SC	20.720	14.093	29.412	18.045	6.517	4.796
RW	22.899	19.576	19.352	11.332	2.757	1.718
RC	23.741	18.553	43.573	15.434	12.749	0.333
JK	39.203	9.794	26.355	11.204	0.711	0.465
gold	21.763	6.462	13.934	6.573	0.179	0.000
CallHome	17.108	13.099	21.962	11.003	5.126	1.932
NewsHour	25.828	16.733	29.536	13.530	4.300	2.947
CrossFire	33.923	22.132	21.705	10.615	1.853	0.976
Meetings	37.674	23.413	23.034	9.222	7.456	1.123
all dialogues	23.152	16.173	22.796	10.807	4.665	1.636

Table 2.2
Nuclei and satellites: length in tokens and relative frequency (in % of all tokens)

between 17.1 (13.1) tokens for CALLHOME and 37.7 (23.4) tokens for GROUP MEETINGS data — this is most likely a reflection on the typical length of turns in the different sub-corpora

- a similar variation is also observed across annotators: between 12 and 40 tokens for nucleus-IUs and between 9 and 20 tokens for satellites — the granularity of IUs is quite different across annotators
- since some annotators are also more greedy in marking IUs than others, there is an even larger discrepancy in the relative number of words assigned to n-IUs and s-IUs among the different annotators: 11-44% (n-IUs) and 0-13% (s-IUs)
- the ratio of nucleus vs. satellite tokens also varies greatly among the annotators: from about 1:1 to 40:1
- the ratio of nucleus- and satellite-tokens which are marked as "indispensable" ("+"-marked) varies greatly: between 36 and 77% for n-IUs and between 2 and 80% for s-IUs

From these observations, we can conjecture that merging the nucleus- and satellite-IUs into one class would yield a more consistent picture than keeping them separate. Also, given this data, we deem it highly unlikely that the agreement just on n-IUs or s-IUs would be larger than on any relevant marked passages in general. As for the "+"-marked passages, we also think that with this high inter-coder variation in relative "+"-marking, it would not make too much sense to keep this distinction in our evaluations.

Further, we conjecture that the (average) length of our extraction units should be in the 10-40 words range (which roughly corresponds to about 3-12 seconds real time, assuming an average word length of 300 milliseconds). Note that (Valenza et al., 1999) found that using 30-grams for summary composition worked well in their experiments which is in line with our observations here on typical human IU lengths.

Example dialogue (`en_4157`)

We will now look at a concrete example of a dialogue which has rather low inter-coder agreement (from English CALLHOME) and analyze the information content and how it was marked by the different human annotators. For this analysis, we take a comparative look at sub-topical passages within this dialogue. We provide a brief characterization of each passage in the following and then show which passages got selected as relevant in Table 2.3.

```
Turns: passage content
 1- 7: about Cynthia / spk.A seeing her
 8-12: about Cynthia's boyfriend Kurth
13-17: about a wedding
18-31: A's trip to Spain and A taking resumes on the trip
32-38: about A missing something and B not [but what this is, does
       not become clear, maybe living in the US?]

39-48: B's fiance got a job offer, so they could go to Switzerland
49-58: B is sceptic on going there (wedding...)
59-66: B's fiance may go there October already

67-73: about B coming back eventually (to US)
74-85: picking up on A's ``missing something'', A ``hates everything''
       but uses her Spanish at work
86-92: B hates her job; her last day is
       Thursday and she actually works only until Tuesday

 93-100: someone is coming the following Saturday
```

```
101-110: about two friends (Ellen, Reuben) and whether
         they are coming (to B?)
111-126: about Erica, her boyfriend and their relationship
127-129: friends of B from France coming
130-147: B's fiance's family: who comes to the wedding and
         whom B likes/dislikes

148-153: B's plans (after the wedding)
154-160: B's moving ordeal
161-166: a friend of B's move

167-171: A's plans (Masters in Madrid, working at TGI Fridays)
172-187: which US universities would have a program; NYU's masters
188-193: Middleburg's masters

194-204: A complains about being bored and still being single
205-211: A talks with her boyfriend about wedding
212-222: A is envious about most of her friends being/getting married
         and B is supportive

223-233: A's boyfriend: his job
234-255: ... that he doesn't consider a long-term relationship
256-270: ... if A might go with him to Spain if he asked her to
271-280: where in Madrid A's boyfriend lives; his parents' jobs
```

Annotator JK's strategy is often to mark a sequence of adjacent turns as a single nucleus-IU, which is not exactly according to the annotation rules. However, it can be argued for that those stretches of information are more coherent than smaller IUs which are more evenly spread across the dialogue. Annotator RC's strategy is in a sense complementary: He marks brief IUs in a pretty regular pace over the dialogue, thus covering almost anything going on to some extent (more than 55% of the tokens are within a nucleus or a satellite, as opposed to 17–36% for the 5 other annotators). To keep the summary length within the prescribed limits, he makes extensive use of "+"-marking within the nucleus-IUs.

We computed mean and standard deviation (s) for each sub-topical passage (29 passages total, only the four annotators, excluding the gold standard annotation; we add both marked and unmarked tokens, since they combined represent an indication of emphasis for a particular passage). We find $s > 5$ for 4 passages, $s > 3$ for 12 passages, and $s < 2$ for only 11 passages. The four passages with the most diverse relevance rankings are: turns 111-126, 130-147, 172-187, and 234-255. To

turns	JK	RC	LB	RW	GOLD
1-7	0.0	1.5 (1.1)	0.9	0.0	1.3 (3.4)
8-12	0.2	0.8 (0.9)	0.2	0.2	0.3
13-17	3.9 (6.8)	0.3 (4.5)	2.7 (2.2)	0.0 (0.2)	0.3
18-31	4.5 (5.0)	1.3 (4.2)	7.4 (2.2)	3.7 (2.4)	4.0 (4.7)
32-38	0.2	0.0 (0.4)	0.2	0.5	0.3
39-48	2.1 (0.6)	2.1 (4.1)	7.2 (2.5)	4.4 (6.0)	7.7 (3.0)
49-58	1.0 (0.3)	1.3 (4.6)	1.1	2.2 (0.5)	0.3
59-66	0.2	0.0 (1.2)	1.3	0.0 (0.2)	0.3
67-73	0.2	1.2 (2.7)	0.2	1.6 (0.5)	2.7 (0.3)
74-85	0.2	1.3 (3.5)	6.5	0.2	3.0 (1.0)
86-92	0.2	0.8 (2.0)	3.6	0.2	1.3 (1.3)
93-100	0.0 (0.2)	0.0 (0.1)	0.2	0.2	0.0 (0.3)
101-110	1.1 (4.5)	0.1	0.2	3.8 (3.7)	3.7 (4.0)
111-126	3.1 (16.2)	1.8 (5.7)	3.1 (0.4)	3.1 (0.9)	3.7 (2.7)
127-129	1.1 (0.5)	0.4 (0.2)	0.2	1.3 (0.7)	0.3
130-147	1.3 (0.6)	1.4 (3.8)	11.5 (2.5)	2.0 (0.2)	0.0 (0.3)
148-153	1.9 (0.3)	0.7 (1.8)	3.1 (4.3)	3.7 (1.6)	2.3
154-160	1.9 (5.5)	0.8 (1.5)	0.2	0.2	1.0 (2.3)
161-166	0.3	0.0 (4.6)	1.8 (2.5)	0.2	1.3 (1.3)
167-171	1.3 (0.3)	1.3 (1.7)	0.2	2.9 (1.1)	1.7 (1.7)
172-187	1.8 (0.5)	1.0 (1.5)	3.4 (1.1)	7.0 (7.5)	2.7 (0.7)
188-193	1.8 (0.5)	1.1 (2.7)	6.3 (1.6)	5.1 (4.0)	2.0 (1.3)
194-204	2.4 (1.6)	2.1 (3.9)	4.5	5.1 (4.6)	5.4 (8.7)
205-211	3.1 (6.3)	0.0 (2.7)	1.3 (0.9)	0.2	0.3
212-222	0.2	1.8 (2.8)	0.2	0.2	0.3
223-233	0.2	1.0 (3.5)	7.2 (0.7)	4.8 (2.2)	3.0 (1.7)
234-255	4.8 (8.1)	2.8 (4.6)	0.2	6.2 (4.6)	3.0 (2.3)
256-270	1.5 (0.6)	1.2 (2.0)	3.8	0.0	4.4 (1.7)
271-280	1.3	0.0	0.0	0.0	0.0

Table 2.3
Relevance annotations in different sub-topical passages of dialogue `en_4157` by different annotators. All numbers are marked words, in percent of all tokens of one annotator; plain numbers refer to marked words within nuclei/satellites, numbers in brackets to unmarked words within nuceli/satellites; horizontal lines indicate topic boundaries [not all of them are exact since they may have been marked within a sub-topical passage].

conclude this section, we will examine these cases in more detail in the following:

1. 111-126 [Erica, her boyfriend, and their relationship]: If we would look at marked n-IUs and s-IUs only, the variation would be much smaller here, but annotators JK and RC chose much larger IU sections in this passage than the two other annotators.

2. 130-147 [wedding guests]: Here the variance is mostly due to annotator LB's strong emphasis on this passage.

3. 172-187 [US graduate programs]: Annotators JK and RC give only moderate emphasis to this passage, whereas LB and RW (in particular) consider it to be much more relevant.

4. 234-255 [A's boyfriend's perspective on relationship]: This is considered to be highly relevant by all annotators except for LB who chooses the previous passage [223-233, A's boyfriend's job] to be the most salient in this topical segment.

2.2.4 Inter-coder agreement

Agreement between coders (and between automatic methods and coders) has been measured in the summarization literature with quite a wide range of methods: e.g., Rath, Resnick, and Savage (1961) use Kendall's τ, Kupiec, Pedersen, and Chen (1995) (among many others) use percent agreement, and Aone, Okurowski, and Gorlinsky (1997) (among others) use the notions of *precision*, *recall*, and F_1*-score*, which are commonly used in the information retrieval community.

Carletta (1996) makes a strong argument to arrive at a more uniform evaluation metric for inter-coder and intra-coder agreement statistics than it has been the case in the field of natural language processing. Carletta proposes to use Cohen's κ (Cohen, 1960) which was devised for exactly that purpose in non-parametric statistics of multinomial data where classes of disjoint categories are assigned to entities by different (subjective or objective) judges. Although Carletta's argument was mainly intended for the discourse analysis community (e.g., agreement on discourse boundary or speech-act assignment), we argue that the κ-metric would also makes sense in the field of relevance assessment, particularly when the classes are not binary (relevant vs. non-relevant), but multi-valued. We note, however, that for the case where we have *rankings* available for an identical set of text passages, Kendall's τ would be more meaningful. The problem with the latter in the context of spoken language summarization, however, is that it would require clearly defined clausal or sentential units which are not available for spoken language. Also, the effort of explicitly ranking every sentence is much greater and the task much harder to accomplish compared to assigning sentences into, say, three distinct classes of relevance.

We use the κ-statistic in its extension for more than two coders (Davies and Fleiss, 1982) for inter-coder agreement with respect to topical boundaries (agreement is found if boundaries fall within the same 50-word bin of a dialogue) and relevance markings (on a word level). For relevance markings, we compute κ both for the 3-way case (nucleus-IUs, satellite-IUs, unmarked) and the 2-way case (any IUs, unmarked).[6] Topical boundary agreement was not evaluated for 2 of the GROUP MEETING dialogues, where only one of 4 annotators marked any boundary.

To compute κ, we use the following equation from Davies and Fleiss (1982):

$$\kappa = 1 - \frac{IJ^2 - \sum_i \sum_c Y_{ic}^2}{I[J(J-1)\sum_c \bar{P}_c(1-\bar{P}_c) + \sum_c \sum_j (P_{jc} - \bar{P}_c)^2]} \tag{2.1}$$

where C=categories in the classification, I=number of data points (words or boundaries) compared, J=number of coders, Y_{ic}=sum over all coders of all classifications of data point i and category c, P_{jc}=proportion of data points assigned to category c by coder j, and $\bar{P}_c$=overall proportion of classifications to category c.

We compute agreements for each dialogue separately and report the arithmetic means for the five sub-corpora in Table 2.4. We observe that agreement for topical boundaries is much higher than for relevance markings. Furthermore, agreement is generally higher for CALLHOME and comparatively low for the GROUP MEETING corpus.

In the system evaluation we take this low inter-coder agreement into account in that we give equal weight to each annotator's opinion and compute an average relevance score for each word of a dialogue, depending on how many annotators marked that word as belonging to a nucleus-IU or a satellite-IU.

[6] These computations were performed for those 4 (out of 6) annotators who completed the entire corpus mark-up.

Table 2.4
Inter-coder annotation κ agreement for topical boundaries and relevance markings.

	8E-CH	4E-CH	NHOUR	XFIRE	G-MTG	Overall
topical boundaries	0.503	0.402	0.256	0.331	0.174	0.384
relevance markings (3 way)	0.147	0.161	0.123	0.089	0.040	0.117
relevance markings (2 way)	0.157	0.169	0.124	0.100	0.046	0.126

Table 2.5
Inter-coder annotation F_1-agreement for topical boundaries and relevance markings.

	8E-CH	4E-CH	NHOUR	XFIRE	G-MTG	Overall
topical boundaries	.54	.44	.53	.38	.18	.45
relevance markings (2 way)	.38	.39	.38	.32	.32	.36

To compare the κ-evaluation to another commonly used metric, we also compute precision, recall and F_1-scores for the same 4 annotators and the same sets of sub-corpora as before.[7] For topical boundaries, a match means that the boundaries fall within +-3 turns of each other, and for relevant words a match means that the two words either are both marked as relevant or not (nucleus or satellite-IUs). The results can be seen in Table 2.5.

As for topical boundaries, we plotted the κ-agreement against the number of topical boundaries per dialogue (again, for 21 dialogues only; 4-coder agreement). As can be seen in Figure 2.3, most of the low agreement numbers stem from dialogues with rather few topical segments and vice versa; the Pearson r correlation coefficient between number of topical segments and κ agreement is $r = .63$ (significant at $p < 0.01$).

2.2.5 Disfluency and sentence boundary annotation

In addition to the annotation for topic boundaries and relevant text spans, the corpus was also annotated for speech disfluencies and sentence boundaries in the same style as the Penn Treebank SWITCHBOARD corpus (LDC, 1999b). One coder (different from the six annotators mentioned before) manually tagged the corpus for disfluencies and sentence boundaries following the SWITCHBOARD disfluency

[7] Precision is the ratio of correctly matched items over all items (boundaries, marked words), recall the ratio of correctly matched items over all items that need to be matched, and the F_1-score combines precision (P) and recall (R) in the following way: $F_1 = \frac{2PR}{P+R}$.

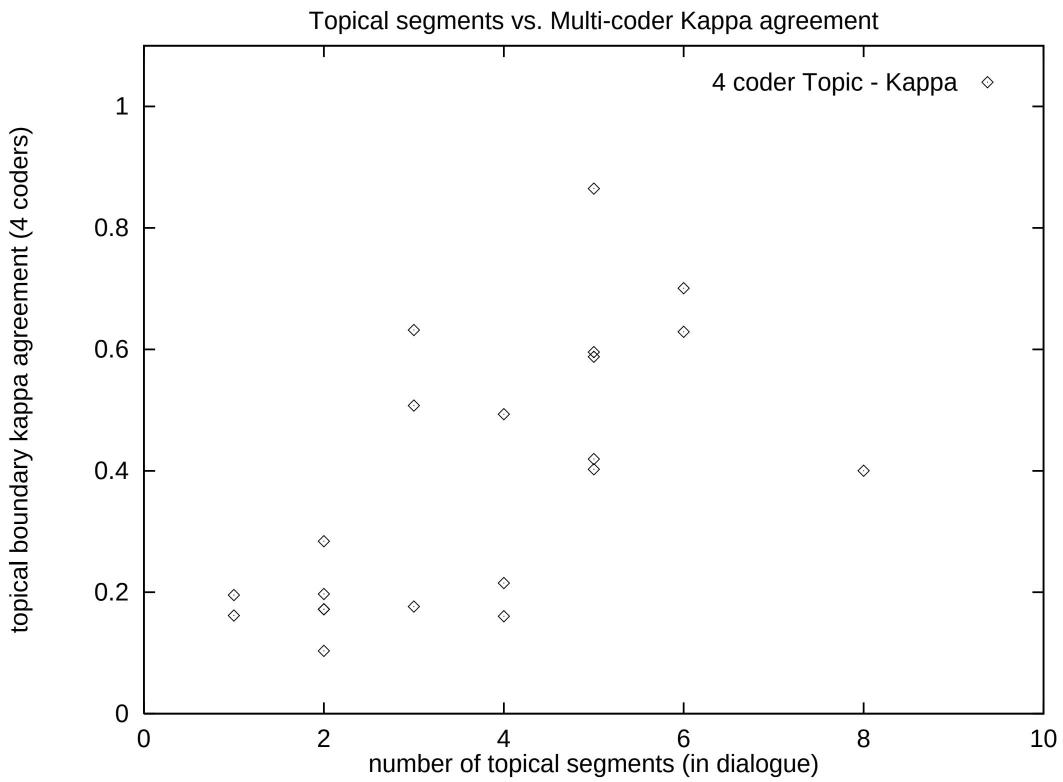

Figure 2.3
Inter-coder κ agreement dependent on the number of topical segments per dialogue.

annotation style book (Meteer et al., 1995). We decided not to annotate ASIDES ({A...}) which are (i) extremely rare and (ii) not always content-less in the first place. Appendix B.2 provides a portion of a file with these disfluency annotations.

2.2.6 Question-Answer annotation

A final type of annotations was performed on the entire corpus to mark all questions and their answers, for the purpose of training and evaluation of the Q-A linking system component. Questions and answers were annotated in the following way: Every sentence which is a question was marked as either a Yes-No-question or a Wh-question. Exceptions were back-channel questions, such as "Is that right?", rhetorical questions, such as "Who would lie in public?", and other questions which do not refer to a propositional content. These were not supposed to be marked (even if they have an apparent answer), since we see the latter class of questions as irrelevant for the purpose of increasing the local coherence within summaries. For each Yes-No-question and Wh-question which has an answer, the answer was marked with its relative offset to the question it belongs to. Some answers are continued over several sentences, but only the core answer (which usually consists of a single sentence) is marked. This decision was made to bias the answer detection module towards brief answers, and to avoid the question-answer regions getting too lengthy, at the expense of summary conciseness.

I shall also call the whole [of language],
consisting of language
and the actions into which it is woven,
'language-game.'
L.Wittgenstein

Chapter 3

Dialogue Summarization System

3.1 System Architecture and Overview

The global system architecture of the spoken dialogue summarization system presented in this thesis (DIASUMM) is depicted in Figure 3.1. The input data is a time ordered sequence of speaker turns with the following quadruple of information: start time, end time, speaker label, and word sequence. The output is the dialogue summary, which, in addition to these four fields of information, is also annotated with topical boundary information, most salient words within a topic, and information about question-answer pairs.

The seven major components are executed sequentially, yielding a pipeline architecture. The sequence was derived from the following contraints:

1. the sentence selection component has to be last, since it needs the complete information from all the other components
2. the topic segmentation module should be as late as possible to be able to have the maximal amount of input available (e.g., sentence boundaries, Q-A-linkages)
3. the POS tagger has to be executed before the other disfluency modules and the Q-A detection component, since they all use POS information

4. sentence boundary detection has to be done before false start and Q-A detection, since both of these components rely on sentence boundary information

5. Q-A detection needs to know about false starts, so it should be run after the false start detection component

6. the repetition filter needs sentence boundary information and thus has to be executed after that module[1]

The following sections describe the components of the system in more detail.

The three components involved in disfluency detection are the part of speech (POS) tagger, the false start detection module, and the repetition filter. They, together with the sentence boundary detection module, are discussed in section 3.3. The question-answer pair detection is described in section 3.4, the topic segmentation in section 3.5, and the sentence selection module, performing relevance ranking, in section 3.6.

3.2 Input Tokenization

We start with a dialogue transcript in textual form, either generated by a human transcriber or by an automatic speech recognition engine. We eliminate all human and non-human noises and incomplete words from the input transcript.[2] Further, we eliminate all information on case and punctuation, since we emulate the ASR output in that regard which does not provide this information. Of course, we are aware that keeping all these features would make the summarization task easier — e.g., we wouldn't need a sentence segmentation module —, but we purposefully did not want to base our results on anything which is due to the fact that the dialogue transcripts resemble written documents in these regards.

Contractions such as `don't` or `I'll` are expanded and treated as separate word tokens — in these examples we would obtain: `do n't` and `I 'll`.

[1]The repetition filter could have been placed somewhat earlier in the pipeline, but we chose to put this module last in the disfluency detection sequence due to details of our implementation.

[2]Most speech recognizers don't output incomplete words, but they do output various noise words; we decided not to consider them here, partly because they were not annotated in the TV shows transcripts.

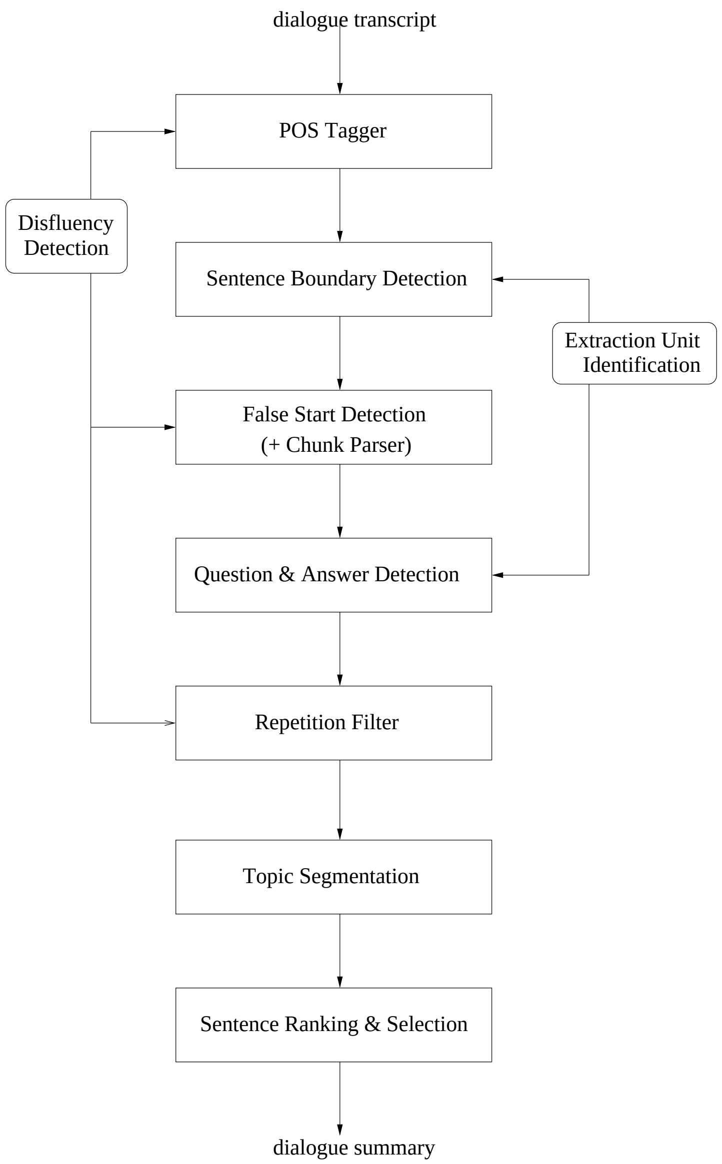

Figure 3.1
Global system architecture.

In Appendix C, we explain and present an example of the internal data format used for the DIASUMM summarization system.

3.3 Disfluency Detection

3.3.1 Motivation

Conversational, informal spoken language is quite different from written language in that a speaker's utterances are typically much less well-formed than a writer's sentences. We can observe a set of disfluencies such as false starts, hesitations, repetitions, filled pauses, and interruptions.

Additionally, in speech there is no good match between linguistically motivated sentence boundaries and turn boundaries or recognition hypotheses from automatic speech recognition.

The following four components can be seen to serve these two main functions of text normalization: (i) disfluency detection, and (ii) sentence boundary detection.

3.3.2 Types of disfluencies

The classification of disfluencies in this thesis follows (Shriberg, 1994; Meteer et al., 1995; Rose, 1998). It is worth noting, however, that any disfluency classification will only be an approximation of the assumed real phenomena and that often boundaries between different classes are fuzzy and hard to decide for human annotators (cf. (Meteer et al., 1995) on annotators' problems with the classification of the word *so*).

- **Filled pauses**: We follow Rose's classification of non-lexicalized filled pauses (typically *uh, um*) and lexicalized filled pauses, e.g. *like, you know* (Rose, 1998). While the former are usually non-ambiguous and hence easy to detect, the latter are ambiguous and much harder to detect accurately.

- **Restarts or repairs**: These are fragments which are resumed, but without completely abandoning the first attempt. We follow the notation in (Meteer et al., 1995; Shriberg, 1994) which has these parts: (a) reparandum, (b) interruption point (+), (c) interregnum (editing phase, {...}), and (d) repair.

- **Repetitions:** A restart with a verbatim repetition of a word or a sequence of words: *[she is + she is] happy*
- **Insertions:** A repetition of the reparandum, with some word(s) inserted: *[she liked + {um} she really liked] it*
- **Substitutions**: The reparandum is not repeated: *[she + {uh} my wife] liked it*

- **False Starts:** These are abandoned, incomplete clauses. In some cases, they may occur at the end of an utterance, and they can be due to interruption by another speaker. (Shriberg (1994) would classify them as *deletion-type repairs*, but we follow Rose (1998) who puts the false starts in a separate class.) Example: ***so we didn't* – / *they have not accepted our proposal***

3.3.3 Related work

The past decade has shown a substantial amount of research in the areas of detecting intonational and linguistic boundaries in conversational speech, as well as detecting and correcting speech disfluencies. While earlier work tended to look at these phenomena in isolation (Nakatani and Hirschberg, 1994; Stolcke and Shriberg, 1996), more recent work attempted to solve several issues within one framework (Heeman and Allen, 1999; Stolcke et al., 1998).

Most approaches use some kind of prosodic information, such as duration of pauses, stress, and pitch contours, and most of them combine this prosodic information with information about word identity and sequence (n-grams, Hidden Markov Models). In the study of Stolcke et al. (1998), the goal was to detect sentence boundaries and a variety of speech disfluencies on a large portion of the SWITCHBOARD corpus. An explicit comparison was made between prosodic and word based models, and the results showed that a n-gram model, enhanced with segmental information about turn boundaries, significantly outperformed the prosodic model. Model combination improved the overall results only to a small extent. These results are encouraging for our word based approaches for disfluency and sentence boundary detection.[3]

3.3.4 Overview

In the following, we will discuss these four main components of the DIASUMM system:

- a POS tagger which tags, in addition to the standard SWITCHBOARD Treebank-3 tag set (LDC, 1999b), also the following disfluent regions or words: (a) coordinating conjunctions which don't serve their usual connective role, but act more as links between subsequent speech-acts of a speaker (e.g., *and then*, we call them *empty coordinating conjunctions* in the thesis); (b) lexicalized filled pauses (labeled as *discourse markers* in the Treebank-3 corpus; e.g. *you know, like*); (c) editing terms within speech repairs (e.g., *I mean*); and (d) non-lexicalized filled pauses (e.g., *uhm*)
- a decision tree which decides on linguistically motivated sentence boundaries, both within a turn and between two turns of the same speaker
- a decision tree (supported by a shallow chunk parser) which decides whether to label a sentence as a false start
- a repetition detection script (for repeated sequences of up to four words)

3.3.5 Training corpus

For training, we use a part of the SWITCHBOARD transcripts which were manually annotated for sentence boundaries, POS, and the following types of disfluent regions (LDC, 1999b):

- {A...}: asides [very rare; we ignore them in our experiments]
- {C...}: empty coordinating conjunctions (e.g., *and then*)
- {D...}: discourse markers (i.e., *lexicalized filled pauses* in our terminology, e.g., *you know*)

[3] As discussed earlier, we excluded the use of prosodic and acoustic information from this present work, with the exception of start and end time information for speaker turns.

Table 3.1
General characteristics of the SWITCHBOARD Treebank-3 corpus.

	count	%
words total	1438359	100.0
non-disfluent words total	1136972	79.0
disfluent words in non-repair sections	163320	11.4
words in repair sections	158163	11.0
unique words in any disfluent section	301387	21.0

Table 3.2
Disfluency characteristics of the SWITCHBOARD Treebank-3 corpus.

type	%	count
all disfl.	100.0	187240
A	0.3	498
C	28.8	53986
D	17.3	32399
E	2.0	3832
F	24.9	46571
repairs	26.7	49954

- {E...}: editing terms (within repairs, e.g., *I mean*)
- {F...}: filled pauses (non-lexicalized, e.g., *uh*, *um*)
- [... + ...]: repairs: the part before the '+' is called reparandum (to be removed), the part after the '+' repair (proper)

Sentence boundaries can be at the end of completed sentences (E_S) or of non-completed sentences, such as false starts or abandoned clauses (N_S). Table 3.1 provides a general statistics of the disfluencies in the SWITCHBOARD corpus, and Table 3.2 breaks it down into the different disfluency types.

3.3.6 POS tagger

We are using Brill's rule based POS tagger (Brill, 1994). Its basic algorithm at run time (after training) can be described as follows:

1. tag every word with its most likely tag, predicting tags of unknown words based on rules

Table 3.3
POS tagger performance (in percent) after a sequence of training steps (reported on training and test sets). Baseline: before any training and application of any rules (i.e., just picking the most likely tag).

data set	purpose	size (words)	baseline	after phase 1	after phase 2	after phase 3
sw2-part1	train/1	250k	86.8	86.8	94.9	95.0
sw2-part2	train/2	250k	84.9	85.6	94.2	95.2
sw2-part3	train/3	250k	85.1	85.8	93.8	94.4
sw4	test	185k	**84.8**	85.4	93.7	**94.1**

2. change every tag according to its right and left context (both words and tags are considered), following a list of rules

For preprocessing, we replace the tags in the regions of {C...}, {D...}, and {E...} with the tags CO, DM, and ET, respectively. The filler-regions {F...} are already tagged with UH from the beginning (they only contain a few different words such as *um* and *uh*). Lines which contain (marked) typographical errors were excluded from the training corpus. We further eliminated all incomplete words (XX tag) and combined multi-words with a GW tag into a single word (hence eliminating the GW tag[4]). The entire resulting new tag set has 42 tags and is listed in Appendix A (for a description of the POS tags used in that database, see (Santorini, 1990; LDC, 1999a)).

To train the POS tagger, we used the dialogues starting with the prefix `sw2*`, comprising about 750,000 word/tag-pairs. We split this corpus in three parts of about 250,000 words each. The first part was used to train the rules for the prediction of the tags of unknown words (*phase 1*), the second and third part were used for the training of the contextual rules (*phase 2* and *phase 3*). The performance after each training step was measured on an independent test set (all `sw4*` dialogues) of about 185,000 word/tag pairs. Table 3.3 shows the tagging performance on the three training sets and the test set after each training step, compared to the baseline of an untrained POS tagger which picks the most likely tag at all times (the tag with the highest relative frequency for a given word). The trained POS tagger's performance on the test set is 94.1% tag accuracy (baseline: 84.8% accuracy).

[4]The sole function of the GW tag is to label words which are considered to be parts of other words but were transcribed separately.

Table 3.4
Precision, recall and F_1-scores of the four disfluency tag categories for the SWITCHBOARD test set

description	count	tag	precision	recall	F_1
empty coord. conjunctions	5990	CO	0.84	0.93	0.88
lexicalized filled pauses	5787	DM	0.95	0.90	0.93
editing terms	1004	ET	0.98	0.94	0.96
non-lexicalized filled pauses	12926	UH	0.98	0.98	0.98

	8E-CH	4E-CH	NHOUR	XFIRE	G-MTG
known words	92.8	90.6	92.7	90.6	93.2
unknown words (total)	48.0 (25)	44.4 (9)	69.6 (23)	86.4 (22)	92.6 (27)
overall	90.6	89.8	91.6	90.4	93.2

Table 3.5
POS tagging accuracy on 5 sub-corpora (evaluated on 500 word samples).

Table 3.4 shows precision, recall, and F_1-scores for the four categories of disfluency tags, measured on the test set after the last training step. We see that the non-lexicalized filler words are almost perfectly tagged ($F_1 = 0.98$), whereas the hardest task for the tagger are the empty coordinating conjunctions ($F_1 = 0.88$): there are a few highly ambiguous words in that set, such as *and, so,* or *or*.

Table 3.5 shows the POS tagging accuracy on the 5 sub-corpora of our dialogue corpus, evaluated on a manually POS tagged sample of 500 words per sub-corpus. We see that the POS tagging accuracy is slightly lower than for the SWITCHBOARD set which was used for training (approx. 90-93%, global average: 91.1%). Further we observe that with the exception of the CALLHOME corpora, the majority of unknown words were actually tagged correctly. The most frequent errors were (a) conjunctions tagged as empty coordinated conjunctions, (b) proper names tagged as regular nouns, and (c) adverbs tagged as adjectives.

Finally, we look at the POS tagger's performance for the four disfluency tags CO, DM, ET, and UH in our 5 sub-corpora; the results of this evaluation are presented in Table 3.6. We can see that the detection accuracy is generally lower than for the corpus we trained the tagger on (SWITCHBOARD), but still quite good in general. The major exceptions are the UH tags where the F_1-scores are comparatively low across the board. The reason for this can be found mostly in words like

	8E-CH	4E-CH	NHOUR	XFIRE	G-MTG
CO	.89	.89	.38	.77	.54
DM	.93	.73	.90	.82	.30
ET	.95	.95	(.94)	.85	.88
UH	.56	.62	(.14)	(.28)	.45

Table 3.6
Disfluency tag detection (F_1) for 5 sub-corpora (Results in brackets: Less than 10 tags to be detected.)

yes, no, uh-huh, right, okay, yeah which are often tagged with UH in SWITCHBOARD but frequently are not considered to be irrelevant words in our corpus and hence not marked as disfluent (e.g., if they are considered to be the answer to a question or a summary relevant acknowledgment). We circumvent potential exclusion from the summary output of these and other words which might be erroneously tagged as non-lexicalized filled pauses (UH) by marking a small set of words as exempt from removal (see 3.4.6).

3.3.7 Sentence boundary detection

The purpose of this component is to insert linguistically meaningful sentence boundaries in the text, given a POS tagged input. We consider all intra-turn and inter-turn boundary positions for every speaker in a conversation. We use the abbreviations EOS for *end of complete sentence* (E_S in the SWITCHBOARD corpus) and NEOS for *end of non-complete sentence* (N_S in the SWITCHBOARD corpus).

The frequency of sentence boundaries (with respect to the total number of words) is about 13.3%, most of the boundaries (almost 90%) being end-markers of completed sentences. If we would mark all inter-word positions with a non-boundary, this would thus yield a baseline error rate of 13.3%.

We use Release 8 of the C4.5 decision tree distribution (Quinlan, 1992). We use a context of 4 words before and after a hypothesized sentence boundary, motivated by the results of Gavaldà, Zechner, and Aist (1997). Also following Gavaldà, Zechner, and Aist (1997), we use 60 trigger words with high predictive potential, using the score computation method described in this paper. The trigger words were obtained using the `sw2`-part of the corpus.

The decision tree input features for every word position are:

- POS tag
- trigger word
- turn boundary before this word?
- if turn boundary: length of pause after last turn of same speaker

We can either encode POS tags and trigger words as attributes with 42 and 63 values, respectively.[5] or as a list of all the POS tags and trigger words with the values `on` or `off`. We call the latter the binary encoding option.

We used the first 10000 words from the `sw3`-part[6] of the corpus for initial explorative training. We created a sequence of words for each conversation side (speakers A and B, respectively). To obtain the inter-turn pause durations, we had to automatically align the forced aligned Switchboard marker-files (`.mrk`) with the disfluency annotated files (`.mgd`) from the Treebank-3 corpus.

We found the binary encoding version performing superior over the multi-valued encoding version. Its disadvantage, however, is its much larger feature space, file size, and training time. We further observed that the very infrequent NEOS boundaries (only about 10% of all boundaries which is only about 1.3% of all potential boundaries) are extremely hard to detect correctly. We therefore merged this class with the EOS class and report results for this combined class only: CEOS. We rely on the false start detection module described below to identify the NEOS sentences within this merged class of sentences.

After these initial exploratory experiments, we used the first 50000 words from the `sw3`-part of the Treebank-3 corpus for the actual decision tree training procedure. We used a test set of about 1000 words and increased the training set size successively from 5000 to 9000, 15000, 25000, and 40000, where performance leveled off.

[5] We need three extra values for the trigger words: nontrig, start, end — the two latter for the cases where we have a start or an end of a conversation side and need padding at either end of this text.

[6] Files starting in `sw3*`.

Table 3.7
Sentence boundary detection accuracy on unseen data for varying sizes of training sets. (Using inter-turn pause length and turn boundary information.)

training set size (examples)	5000	9000	15000	25000	40000
CEOS (F_1)	.841	.851	.860	.887	.892
non boundary (F_1)	.971	.970	.972	.977	.978
classification error (%)	4.9	4.9	4.7	3.8	3.6

Table 3.8
Sentence boundary detection accuracy on unseen data for the training set with 25000 examples (F_1-score)

with inter-turn pause duration?	yes		no	
with turn-boundary info?	yes	no	yes	no
training set	.904	.903	.900	.884
test set	.887	.884	.884	.825

Results show that for good performance we need to know about either one of these two features: "is there a turn boundary before this word?" or "pause duration after last turn from same speaker".

Table 3.7 shows the results of CEOS detection for the varying training sizes, and Table 3.8 shows the results in detail for the various parameter combinations. We note that both in terms of error rate as well as in terms of F_1-scores, our results are well in line with those of Gavaldà, Zechner, and Aist (1997). (They cannot be compared exactly, however, since we are here interested in *sentence* as opposed to *short clause* boundaries.)

Effect of imperfect POS tagging

To see how much influence an imperfect POS tagging might have on these results, we POS tagged the test set data using the POS tagger described in the previous section. While the POS tagger accuracy for this test set was about 95.3%, the F_1-score for CEOS was still .882 (which is 98.9% of .892). This is encouraging since it shows that the decision tree is not very sensitive to the majority of POS errors.

Table 3.9
Inter- and intra-turn boundary detection results on two test sets. (Set 1: 1000 examples, set 2: 10000 examples.)

	occurrence (%)	detection accuracy (F_1)
set 1 : inter-turn non-bd	12 (1.2)	.56
set 1 : inter-turn bd	112 (11.3)	.95
set 1 : intra-turn non-bd	809 (81.4)	.99
set 1 : intra-turn bd	61 (6.1)	.77
set 2 : inter-turn non-bd	96 (1.0)	.60
set 2 : inter-turn bd	676 (6.8)	.94
set 2 : intra-turn non-bd	8588 (86.2)	.98
set 2 : intra-turn bd	608 (6.1)	.72

Inter-turn and intra-turn boundaries

It might also be interesting to consider how the detection of sentence boundaries *between turns* (inter-turn) compares to the detection of boundaries *within a turn* (intra-turn). For this, we use the best decision tree of the above experiments (40000 examples in the training set, using turn boundary and inter-turn pause duration information). We test the accuracy on (a) the test set used above (1000 examples, set 1) and (b) an extended unseen test set of 10000 examples (set 2). The second test set had somewhat lower F_1-scores overall (CEOS: .845; NONE: .978; overall errors: 3.8%). Table 3.9 shows the results of these experiments. As could be expected, the performance is very good for the 2 frequent classes: sentence boundaries at the end of turns and non-boundaries within turns ($F_1 > .94$), but considerably worse for the two more infrequent cases. The very rare cases (around 1% only) of non-sentence boundaries at the end of turns (i.e., turn-continuations) has the lowest performance ($F_1 \approx .6$).

Performance issues

While the best decision tree configuration (binary encoding, trained on 40000 samples) is quite accurate, the problem with that in an actual working system is its performance: due to the binary encoding, both the encoding process and the decision tree run-time are quite long.

To speed up this component in a time-critical run-time system (in particular, the

Meeting Browser), we use (optionally) a decision tree which has non-binary, i.e., multi-valued encoding. On the same test set, the error rate increases from 3.6% to 4.9%, and the CEOS-F_1-score decreases from .892 to .848, recall being more affected than precision (-.063 vs. -.018).

A time-comparison yielded the following results:[7] It took about 480 seconds to encode about 40000 samples as opposed to only 75 seconds if in multi-valued mode (83 samples/sec vs. 533 samples/sec). The decision tree run-time performance can be reduced from about 14 seconds for 10000 test samples to about 2 seconds in multi-valued mode (714 vs. 5000 samples/second). If we have a typical dialogue of 10 minutes length with about 2000 words, the overall time would be reduced from about 24+3=27 to about 4+0.5=4.5 seconds (a six-fold speed-up).

We further tried if the "-s" flag in C4.5 (which allows the user to combine features in groups) could increase the decision tree performance with the multi-valued approach. (While training of this takes much longer, there is no time-effect in encoding and running the resulting decision tree.) Indeed, the error rate is only 4.3% and the CEOS-F_1 .872. So we use this decision tree for the "fast" mode in the DIASUMM system. This variant yields a good compromise between speed and accuracy.

Sentence boundary detection on dialogue corpus

To get a picture about the realistic performance of this component, using the (imperfect) POS tagger and the "fast" decision tree version (see above), we evaluate the sentence boundary detection accuracy for all 5 sub-corpora of our dialogue corpus. Table 3.10 yields the results of these experiments. The results reflect a trend very similar to the SWITCHBOARD corpus, in that the two more frequent classes (inter-turn boundaries and intra-turn non-boundaries) have high detection scores, whereas the two more infrequent classes perform less well. Furthermore, we observe that in cases where the relative frequency of rare classes is further reduced, the classification accuracy declines over-proportionally (particularly for the rarest class of the inter-turn non-boundaries). Also, overall boundary detection is better for the two more informal corpora, CALLHOME and GROUP MEETINGS ($F_1 > .72$).

[7] All experiments were performed on a 167 MHz 320MB Sun Ultra1 workstation.

Table 3.10
Boundary detection accuracy (F_1) for 5 sub-corpora (in brackets: relative frequency of class in percent).

	8E-CH	4E-CH	NHOUR	XFIRE	G-MTG
inter-turn non-bd	.51 (2.9)	.31 (1.4)	[0] (0.0)	.10 (0.1)	.06 (0.1)
inter-turn bd	.84 (9.9)	.89 (12.3)	.93 (2.0)	.89 (2.9)	.93 (5.4)
intra-turn non-bd	.97 (80.7)	.97 (79.5)	.97 (91.8)	.97 (91.2)	.97 (87.6)
intra-turn bd	.60 (6.5)	.65 (6.8)	.56 (6.2)	.42 (5.8)	.56 (6.9)
overall bd	.75 (16.4)	.80 (19.1)	.66 (8.2)	.58 (8.7)	.72 (12.4)
overall non-bd	.95 (83.6)	.96 (80.9)	.97 (91.8)	.97 (91.3)	.96 (87.6)

3.3.8 Repetition detection

This component is concerned with (verbatim) repetitions within a speaker's turn, the most frequently occurring case of all speech repairs for informal dialogues (insertions and substitutions are comparatively less frequent). Repeated phrases can be potentially interrupted by other disfluencies, such as filled pauses or editing terms. Repetition detection is done with a script which can identify repetitions of word/POS sequences of length 1 to 4 (longer repetitions are extremely rare: on average, less than 1% of all repetitions listed in Table 3.11). Words which were marked as disfluent by the POS tagger are ignored when considering the repeated sequences, so we can correctly detect repetitions like: [he said uh to + he said to] him...

Table 3.11 provides the relative frequency of different types of repairs, both for a subset the SWITCHBOARD corpus (about 933,000 words), as well as for the 5 sub-corpora of our database (about 47,000 words total). (We say *multiple repetitions*, if a sequence of words is repeated more than once, and *single repetitions* otherwise.) If we assume to have perfect POS tags, as well as correct sentence boundaries, our repetition script can correctly identify about 30–50% of all repairs in the informal sub-corpora, including SWITCHBOARD (classes A, B, and C in the table).

Finally, we are evaluating the precision, recall, and F_1-scores for this component at the level of individual words when using the POS tagger and the sentence boundary detection component. Table 3.12 shows the results. We see that for the informal sub-corpora, we get very good precision (only few repetitions detected are incorrect) and recall is in the 25-45% range (since we cannot detect substitution

	SWBD	8E-CH	4E-CH	NHOUR	XFIRE	G-MTG
repairs total (=100%)	22725	414	167	20	61	273
(A) mult. repetitions	2.9	3.1	—	—	—	7.3
(B) single repet. without disfl.	37.6	32.4	25.7	10.0	13.1	40.7
(C) single repet. with disfl.	6.2	4.3	4.8	—	—	3.7
(D) other types of repairs	53.3	60.1	69.5	90.0	86.9	48.4

Table 3.11
Relative frequencies in % of various types of repairs in different corpora.

	8E-CH	4E-CH	NHOUR	XFIRE	G-MTG
repair tokens (%)	4.7	3.8	2.2	1.3	7.9
precision	.88	.78	.25	.35	.91
recall	.41	.32	.01	.04	.27
F_1-score	.56	.45	.02	.08	.41

Table 3.12
Detection accuracy for repairs on the basis of individual word tokens using the repetition filter.

or insertion type of repairs). The results for the formal sub-corpora are considerably worse, so this filter should probably not be used for corpora with as few repetitions as NEWSHOUR or CROSSFIRE. We checked all of the 95 *false positives* of this evaluation and observed that in the majority of cases (41%), the repetition was correctly detected, but was not marked by the human annotator, since it might be considered as a case of emphasis. We believe that while some nuances of the sentence(s) might be lost, for the purpose of summarization it makes perfect sense to reduce this information. Sometimes, individual words are repeated for emphasis, sometimes sentences, e.g., "Good./ Good./". We provide an example, where this emphasis is extreme (from a CALLHOME dialogue: `en_5648`):

```
203  776.59 778.72 B: [...] How is the new person doing? q/
204  779.25 782.33 A: Very very very very very well. / [...]
```

Further, about 19% of false positives were correct but not annotated since they span multiple turns, and about 14% were erroneously missed by the human annotator. Only the remainder of cases (26%) were actual false positives, caused by incorrect POS tags (5%, typically an incorrectly tagged "that/WDT that/DT" sequence at the beginning of a relative clause) or missing sentence boundaries (21%).

There have been attempts to get a more complete coverage of detection and correction of all types of speech repairs (Heeman and Allen, 1999). However, we decided to use a simple method here which works well for a large subset of cases and is very efficient at the same time.

3.3.9 False start detection

False starts are quite frequent in spontaneous speech, occurring at a rate of about 10-15% of all sentences (SWITCHBOARD, CALLHOME). They involve less than 10% of the total words of a dialogue; about 34% of the words in these incomplete sentences are part of some other disfluencies, such as filled pauses or repairs. (In complete sentences, only about 15% of the words are part of some disfluencies.) For CALLHOME, the average length of complete sentences is about 6 words, of incomplete sentences about 4.1 words (including disfluencies).

We trained a decision tree on 8000 sentences of SWITCHBOARD. As features we use the first and last four trigger words[8] and POS of every sentence, as well as the first and last four chunks from a POS based chunk parser. This chunk parser is based on a simple context-free POS grammar for English and uses a heuristic for maximal coverage of the input for ambiguity resolution. Its output are common phrases such as noun phrases or prepositional phrases. We only slightly modified the chunk grammar from Zechner (1997); in the evaluation system (on the Sun platform), we use the PHOENIX parser (Ward, 1991), for the Meeting Browser server (on the NT platform), we use a version of the SOUP Parser (Gavaldà, 2000).[9] Further, we encode the length of the sentence in words and the number of the words not parsed by the chunk parser.

Since binary encoding did not turn out to be better than non-binary encoding, we stayed with the latter encoding which is also faster in encoding time and run time due to a much smaller feature space. Further, we observed that while the chunk information itself does not really improve performance over the baseline of using trigger words and POS information only, the derived feature of “number of not parsed words” actually does improve the results.

We ran the decision tree on data with perfect POS tags (for SWITCHBOARD only), disfluency tags (except for repairs) and boundaries. For the SWITCHBOARD

[8] We used the same trigger words as for the sentence boundary detection component

[9] Their output is equivalent for the purposes of this thesis.

corpus, we ran a version where disfluencies were removed prior to parsing, another where they were kept. The evaluations were performed on independent test sets of about 3000 sentences for SWITCHBOARD and of our complete dialogue corpus. Table 3.13 shows the results of these experiments. We see that keeping disfluencies yields better results; only about 2% of the complete sentences are incorrectly classified as incomplete. However, on manual inspection, we see that some of those sentences are quite long and there is some danger of losing too much information by simply removing them right away. Typical errors, where complete sentences were classified as incomplete, are inverted forms or ellipsis at the end of a sentence, e.g. *neither do I*, *it seems to*. The performance for the informal corpora (CALLHOME, GROUP MEETINGS) is better than for the formal corpora (NEWSHOUR, CROSSFIRE); this is related to the fact that the relative frequency of false starts is markedly lower in these latter data sets.

3.3.10 Disfluency correction

After detection, the correction of disfluencies is straightforward. We eliminate the words that were tagged with CO, DM, ET, or UH by the POS-Tagger (possibly except for words like *yeah, yes, no* which might play an important role in Q-A pairs). Further, we remove the false start sections and all initial parts of repetitive sections, e.g., in *[he makes + he makes] good food*, we would remove the first two words.

3.4 Cross-speaker Information Linking

3.4.1 Overview

One of the properties of multi-party dialogues is that shared information is created between dialogue participants. The most obvious interactions of this kind are question-answer pairs. The purpose of this component is to automatically create such coherent pieces of relevant information which can then be extracted together while generating the summary. The effects of such linkings on actual summaries can be seen in two dimensions: (i) increased local coherence in the summary; (ii) a potentially higher informativeness of the summary. Since Q-A linking has a side effect in that *other* information will be lost with respect to a summary of the same

Table 3.13
False start decision tree classifier results for different corpora (NEOS=incomplete sentence=false start; EOS=complete sentence)

Sentences	Precision	Recall	F_1-score
3199	SWITCHBOARD-NO-DISFL		
12.3% NEOS	0.769	0.455	0.571
87.7% EOS	0.923	0.980	0.951
3199	SWITCHBOARD-WITH-DISFL		
12.3% NEOS	0.808	0.491	0.611
87.7% EOS	0.932	0.984	0.957
2212	8CH-DEVTEST		
12.2% NEOS	0.676	0.457	0.545
87.8% EOS	0.928	0.970	0.948
1458	4CH-EVAL		
11.0% NEOS	0.754	0.556	0.640
89.0% EOS	0.947	0.978	0.962
303	NEWSHOUR		
2.0% NEOS	1.000	0.167	0.286
98.0% EOS	0.983	1.000	0.992
1102	CROSSFIRE		
7.4% NEOS	0.704	0.235	0.352
92.7% EOS	0.942	0.992	0.967
1164	GROUP MEETINGS		
13.9% NEOS	0.730	0.451	0.557
86.1% EOS	0.916	0.973	0.944

length without Q-A linking, the second claim is much less certain to hold than the first. We will investigate these questions further and evaluate them in more detail in a later section of this thesis (section 4.4; see also our related paper on this topic (Zechner and Lavie, 2001)). Here we will be concerned only with the following two intuitive sub-tasks of Q-A linking: (i) identifying questions; (ii) finding their corresponding answers.

3.4.2 Related work

Detecting a question and its corresponding answer can be seen as a sub task of the speech-act detection and classification task. Recently, Stolcke et al. (2000) presented a comprehensive approach to dialogue act modeling with statistical techniques. A good overview and comparison of recent related work can also be found in this article. Results from their evaluations on SWITCHBOARD data show that word based speech act classifiers usually perform better than prosody based classifiers, but that a model combination of the two approaches can yield to an improvement in classification accuracy.

3.4.3 Corpus statistics

For training of the question detection module, we used the manually annotated set of roughly 200,000 SWITCHBOARD speech acts[10] (SAs);[11] for training of the answer detection component, we used eight English CALLHOME dialogues (8E-CH), which were manually annotated for question-answer pairs. While we were aiming to detect all questions in the question detection module, the answer detection module focuses on Q-A pairs only: we exclude all questions from consideration which are not Yes-No- or Wh-questions (such as rhetorical or back-channel questions), as well as those which do not have an answer in the dialogue. Further, some questions were answered not in the immediately following turn (*delayed answer*). Table 3.14 provides the frequencies of these events in the 8E-CH-corpus.

[10]In this thesis, the notions of *speech acts* and *sentences* can be considered equivalent.

[11]From the Johns Hopkins University LVCSR Summer Workshop 1997. Thanks to Klaus Ries for providing the data. The data is also available from `http://www.colorado.edu/ling/jurafsky/ws97/`.

Table 3.14
Frequency of different types of questions in the 8-English-CALLHOME data set.

turns	2211
Wh-questions total	20
... with immediate answers	15 (75%)
YN-questions total	48
... with immediate answers	38 (79%)
Qs excluded for Q-A detection	15
questions total	83 (3.75%)

3.4.4 Automatic question detection

We used two different methods: (a) a speech-act tagger, trained on SWITCHBOARD data[12] and (b) a decision tree based on trigger word and part-of-speech information.

Speech act tagger

The speech-act tagger tags one speech act at a time and hence can only make use of speech act unigram information. Within a speech act, it uses a language model based on POS and the 500 most frequent word/POS pairs. It was trained on the SWITCHBOARD speech act training set (about 200,000 speech acts). It was not optimized for the task of question detection. Its typical runtime for speech act classification is about 10 speech acts per second.

Decision Tree question classifier

The decision tree classifier (C4.5) uses the following set of features: (a) POS and trigger word information for the first and last five tokens of each speech act[13]; (b) SA length; and (c) occurrence of POS bigrams. The set of trigger words is the same as for the components in the disfluency detection modules. The POS bigrams were designed to be most discriminative between q-SAs (question speech acts) vs. non-q-SAs (non-question speech acts). The bigrams were obtained as follows:

[12] Thanks to Klaus Ries for providing us with the software.

[13] Shorter SAs are padded with dummies.

Table 3.15
q-SA frequencies for the 2 decision tree training sets (questions other than YN/Wh-questions were all mapped to `qother`).

	unbalanced set	balanced set
YN-questions (qy)	539	5569
Wh-questions (qw)	199	1989
other questions (qother)	178	1621
questions total	916 (4.6%)	9179 (50.3%)
non-q SAs (nonq)	18784	9064
total SAs	19700	18243

1. for a balanced set of q-SAs and non-q-SAs (about 9000 SAs each): count all the POS bigrams in positions $1..5$ and $(n-4)..n$ (using START and END for the first and last bigrams, respectively) and memorize position (beginning or end of SA) and type (q-SA vs. non-q-SA)
2. for all bigrams:
   (a) add 1 to the count (to prevent division by zero)
   (b) divide the q-SA-count by the non-q-SA-count
   (c) if the ratio is smaller than 1, invert it (ratio:=1/ratio)
   (d) multiply the ratio with the total frequency of q-SA-count and non-q-SA-count combined[14]
3. extract the 100 bigrams with the highest value

We trained two versions of the decision tree: (a) with an unbalanced training set of about 20,000 SAs from the SWITCHBOARD training data which reflects the true distribution of SAs in general and questions in particular; (b) a balanced training set of about 18,000 SAs from the SWITCHBOARD training data which contains approximately the same number of questions and answers. The motivation for the latter decision tree was to enforce focus on the relatively infrequent Q-classes (see Table 3.15) and hence trying to boost recall at the expense of precision, since the classifier would overestimate the Q-classes on a non-skewed test set.

[14]Leaving out this step favors low frequency high discriminative bigrams too much and causes a slight reduction in overall q-detection performance.

Table 3.16
Question detection on the 8E-CH corpus using three different methods.

	SA tagger	unbalanced DTree	balanced DTree
overall error	3.2%	4.7%	12.6%
q-precision	.57	.63	.26
q-recall	.61	.51	.84
q-F_1	.59	.56	.40
q-pr_{avg}	.59	.57	.55
typical classification time (SAs/sec)	10	1000	1000

Experiments and results

We evaluated the speech act tagger and the decision tree classifiers on the 8E-CH data set which was manually annotated for questions (and their corresponding answers). Whereas in the later stage of answer detection, non-propositional questions are ignored, at this point we are interested in the detection of *all* the kinds of annotated questions in the corpus. This also reflects the fact that the training set contains all possible types of questions. We mapped the SA-hypotheses of both classifiers to the tokens Q and NONQ and then evaluated the precision, recall, and F_1-score for these two main classes.

Since the F_1-score is a parabolic curve, favoring a balance between precision and recall, it is to be expected that the F_1-results for the balanced tree look worse than they actually are: the lower precision score causes an over-proportional drop in F_1-score. We therefore report also another score which better reflects and states the *combined* quality of precision and recall, particularly in a context where the sum of precision and recall is fairly constant and either one of them can be increased or decreased at the expense of the other: $pr_{avg} = \frac{P+R}{2}$.

Table 3.16 reports the results of the question detection experiments with the three different classifiers used, all on the same data set (8E-CH). Note that while the decision trees are performing only slightly worse than the speech act tagger, their typical classification time is two orders of magnitude faster. We use that as an argument to use the two versions of the decision tree in DIASUMM, but not the SA tagger.

3.4.5 Detecting the answers

After identifying which sentences are questions, the next step is to identify their answers. From the 8E-CH-statistics of Table 3.14 we observe that for more than 75% of the YN- and Wh-questions, the answer is to be found in the first sentence of the speaker talking after the speaker uttering the question. In the remainder of cases, the majority of answers are in the second (instead of the first) sentence of the other speaker. Further, there are usually no (or only very few) sentences uttered by the speaker who posed a question *after* the question is being asked and before the next speaker starts talking.

In addition to detecting sequential Q-A pairs, we also want to be able to detect simple embedded questions, as shown in this example:

```
q  1  A: When are we meeting then?
q   2 B: You mean tomorrow?
    3 A: Yes.
   4  B: At 4pm.
```

We devise the following heuristics to detect answers:

- if the first speaker change after the question occurs more than $maxChg$ sentences after the question, the search is stopped and no Q-A-pair is returned
- answer hypotheses are sought for maximally $maxSeek$ sentences after the first speaker change after the question, but not over interruptions by any other speaker, i.e., we check within a single speaker region (this is the stopping criterion for the following two heuristics) — an exception occurs if there is an embedded question in the first single speaker region: in that case, we look at the next region where a speaker different from the Q-speaker is active[15]
- answers have to be minimally $minAns$ words long; if they are shorter, we add the next sentence to the current answer hypothesis
- even if the minimum answer length is reached, the answer can be (optionally) *extended* if at least one word in the answer matches a word from the question (two different stop lists ($StopShort$, $StopLong$), or no stop list are used to

[15]This would be sentence 4 in the example above.

remove function words from consideration)[16]

We have these further restrictions for the case of embedded questions:

1. If we detect a potential embedded Q-A pair, the answer to the surrounding question must immediately follow the answer to the embedded question. (I.e., the region following the potential answer region of the embedded question — sentence 4 in our above example — must (a) not contain a question itself *and* (b) must be from a different speaker than the surrounding question.)

2. A *crossover* is prohibited, i.e., we eliminate all pairs $< Q_j, A_l >$ when a pair $< Q_i, A_k >$ was already detected with $i < j < k < l$ (k, l being start indices of answer spans).

The output of the algorithm is a list of triples $< Q, A_{start}, A_{end} >$, where Q is the sentence-ID of the question, A_{start} the first, and A_{end} the last sentence of the answer. As mentioned above, we only use 68 of the 83 questions marked in the 8E-CH data set for these evaluations, since only these are YN- or Wh-questions that actually *have* answers in the dialogue. There are four possible outcomes for each triple: (a) irrelevant: a Q-A pair with a wrongfully hypothesized question (this is the fault of the question detection module, not of this heuristic); (b) missed: the answer was missed entirely; (c) completely correct: A_{end} coincides with the correct answer sentence ID; and (d) correct range: the answer is contained in the interval $[A_{start}, A_{end}]$ but does not coincide with A_{end}.

For the calculation of precision, recall, and F_1-score, we count classes (c) and (d) as correct and use the sum of all classes for the denominator of precision and the total number of Q-A pairs (68 in this development set) as the denominator of recall.

To determine the best parameters, we varied them across a reasonable set of values and ran the answer detection script for all combinations of parameters. The best results (wrt. F_1-score) using questions detected by the speech act tagger and the two decision trees are reported in Table 3.17. Again, we report pr_{avg}-scores in addition to the F_1-scores.

We make the following observations:

[16] $StopLong$ contains 571 words, $StopShort$ only 89 words, most of which are auxiliary verbs and filler words.

Table 3.17
Q-A-detection results using three different question detection methods (68 Q-A pairs to be detected).

	SA tagger	unbalanced DTree	balanced DTree
all hypothesized Q-A pairs	80	54	173
correct [(c) and (d)]	42	31	44
$maxChg$ (1-5)	4	2	2
$maxSeek$ (2-4)	3-4	2-4	4
$minAns$ (1-10)	5-10	2-10	5-10
similarity extension (on/off)	on	on	on
stop list (no/short/long)	no/short	no/short	no/short
precision	.53	.57	.25
recall	.62	.46	.65
F_1-score	.57	.51	.37
pr_{avg}	.57	.51	.45

- the performance measured with pr_{avg} is about 82-97% of the Q-detection performance;
- on the Q-A detection task, the unbalanced tree performs relatively better than the balanced tree ($pr_{avg,q-a}/pr_{avg,q}$);
- the unbalanced decision tree is a little less sensitive to the parameter setting than the balanced decision tree.

Since it is possible to create the intersection of the best parameter settings for the two decision tree methods, we use that optimal setting in DIASUMM: $maxChg = 2, maxSeek = 4, minAns = 10, sim = on, stop = no$.

Finally, we evaluated the performance of both the Q-detection module and the combined Q-A detection on all 5 sub-corpora, using the unbalanced decision tree for question detection; the results are reported in Table 3.18. Except for the rather small NEWSHOUR corpus (with fewer than 20 questions or Q-A pairs to identify), the typical Q-detection F_1-score is around .6 and the Q-A-F_1-score around .5. In two cases, the Q-A detection performance is slightly better than the Q-detection performance. This can be explained by the fact that the answer detection algorithm prunes away a number of Q-hypotheses, reducing the space for potential Q-A-hypotheses.

	8E-CH	4E-CH	NHOUR	XFIRE	G-MTG
Q to detect	83	94	19	110	49
Q hypotheses	67	60	16	71	52
Q detection (F_1)	.56	.58	.80	.60	.59
Q-A pairs to detect	68	69	18	79	32
Q-A pair hypotheses	54	54	14	54	33
Q-A detection (F_1)	.51	.60	.81	.51	.51

Table 3.18
Performance comparison for Q and Q-A detection (Q detection with unbalanced decision tree).

3.4.6 Q-A detection within DIASUMM

When we use the Q-A detection module in DIASUMM, we want to ensure that (a) there are no Q-A pairs built around Q-sentences which are false starts, and (b) the initial part of an answer is not lost in case the disfluency detection component marks some indicative words as disfluencies. We create a list of such important words (for YN-questions) which are not removed by the summary generator if they appear in the beginning (leading five words) of answers.[17]

3.5 Topic Segmentation

Segmenting text into passages or segments belonging to a single topic can serve a variety of purposes. For instance, in document indexing and retrieval, one can index and retrieve parts of documents instead of whole documents and so focus more on one's particular interests. When summarizing text, be it written or spoken language, topic segmentation can be the first step in gaining a brief gist on the content of a text, in that a list of the most salient keywords within each topical segment can be presented to the user as a first summary approximation. Particularly, in an interactive setting, the user then can choose a subset of topics to be expanded into textual extract summaries, thus performing an interactive "drill-down" summa-

[17]The current list comprises the following words: 'no', 'yes', 'yeah', 'yep', 'sure', 'uh-huh', 'mhm', 'nope'.

rization. Another benefit of having topical segments instead of full texts as input for summarizers lies in the simple fact that their length is more limited, allowing for faster computation in cases of algorithms with a polynomial (non-linear) time behavior (such as MMR). Further, our experiments reported in section 4.2 indicate that segment-based summaries are more accurate than summaries based on the entire dialogue.

In DIASUMM, the topic segmentation module is an integral part of the system. However, for reasons related to evaluation, we do not take into account the performance of this component in measuring overall system performance.[18] The reason for this decision is mostly related to the fact that the human annotators had to mark summary relevant passages on topically coherent regions and not based on the entire dialogues.

3.5.1 Related work

In this thesis, we use an adapted implementation of Hearst's TextTiling approach to automatically find boundaries between topically coherent segments in a text (Hearst, 1997). In the summarization literature, this has been used also by Boguraev and Kennedy (1997) and Barzilay and Elhadad (1997). TextTiling is based on the assumption that topicality is a function of repetitions of words (or their stems) in a text, and that topical shifts are marked by a "break" in the continuity of several strands of these repetitions. Thus, if one observes a large number of words common to two adjacent blocks of text, it is likely that these blocks will belong to the same topic.

Other approaches for topic segmentation are (Morris and Hirst, 1991)'s thesaurus-based *lexical chains* (extensions of chains correspond to extensions of topics), (Youmans, 1991)'s *vocabulary introduction method* (boundaries occur before paragraphs with many new words), or (Kozima, 1993)'s *lexical cohesion profile*, which uses a semantic network to establish text regions of higher similarity. (Hirschberg and Nakatani, 1998) use prosody, and (Litman and Passonneau, 1995) as well as more recently (Shriberg et al., 2000) use, in addition to this, also other linguistic features to segment dialogues. (van Mulbregt et al., 1998) segment texts using a Hidden Markov

[18] Almost all of our evaluations, except for those in section 4.2, are based on pre-determined topical segments, drawn from the human gold standard annotation.

Model, where the states correspond to (automatically clustered) topics. More recently, a purely statistically based approach was introduced by (Beeferman, Berger, and Lafferty, 1999): They propose to segment texts (actually: to find *story* boundaries in Broadcast News data) by incrementally building an exponential model which predicts boundary positions between each pair of words. (Choi, 2000) presented an approach for text segmentation which relies on automatic clustering of individual sentences.

3.5.2 TextTiling

The basic idea of TextTiling which we implemented for the topic segmentation module of DIASUMM is that coherence is a function of repetitions of words (or: their stems) within a window of text: If two adjacent blocks of text share many words, they would belong to the same segment, whereas if their words are (nearly) disjoint, a boundary (or: topic shift) should be assumed in between.

The algorithm (blocks version), which we only slightly modified for our purposes, works as follows (see (Hearst, 1997), .48ff.):

1. Tokenization: convert the text to lower-case tokens, apply a stop list for eliminating non-content words, apply stemming, and use equal-sized groupings of k input tokens: for stemming, we use a simple 6-character truncation, and we experimented with five different stop lists (see section 3.5.4)

2. Computing Scores: the block-similarity score is calculated by a normalized inner vector product of two adjacent blocks' word vectors ($b1, b2$, the w are the frequencies of the stemmed content words within each block):[19]

$$sim(b1, b2) = \frac{\sum w_{b1} w_{b2}}{\sqrt{\sum w_{b1}^2 \sum w_{b2}^2}} \tag{3.1}$$

3. Boundary Identification: The block similarity scores are computed every d words (sample distance) and considered to form a *similarity-graph*, where the potential boundary positions (between two adjacent blocks of text, in our case: between every token in the dialogue) are plotted on the x-axis, and the similarity scores on the y-axis. This similarity-graph is smoothed with a

[19]Since this score is normalized, it is always in the interval [0.0–1.0].

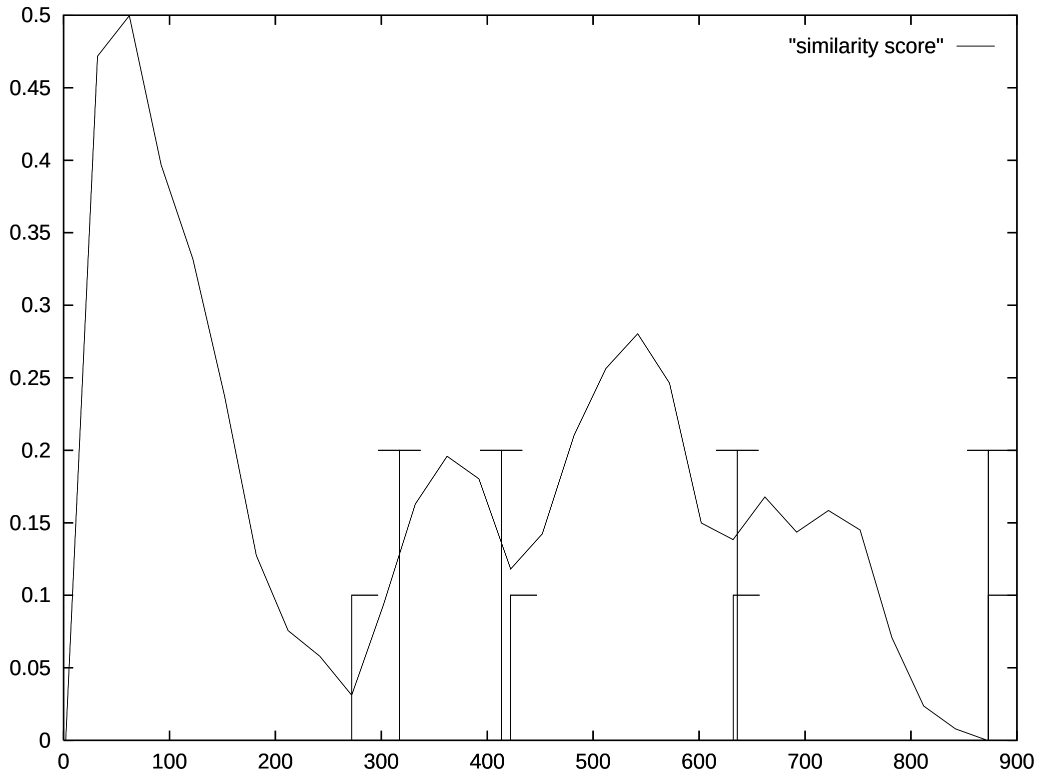

Figure 3.2
Topic segmentation example from an English CALLHOME dialogue.

simple low-pass filter algorithm (averaging over a small window of scores). Then, for each local minimum in the similarity graph, a *relative depth score* d_s is computed which favors "deep valleys" over more "shallow" ones: $d_s = (p_l - m) + (p_r - m)$, where m is the score of the local minimum, p_l is the first local maximum to its left and p_r to its right. Finally, the algorithm assumes boundaries for those local minima whose depth scores exceed $\bar{s} - \alpha\sigma$, $\bar{s}$ being the average depth score, σ the standard deviation and α a tunable parameter.

Figure 3.2 shows an example of the similarity graph for an English CALLHOME dialogue; the shorter vertical lines with the hook to the right indicate the boundaries which were proposed by the algorithm, the longer vertical lines the manually annotated topic boundaries (*gold standard*).

3.5.3 Topic boundary gold standard

The 23 dialogue excerpts in our corpus were annotated for topical boundaries by four to six human annotators. Boundaries were restricted to occur between speaker turns only. As it was expected, the average number of topic boundaries varied between different annotators: For the four annotators completing the entire corpus annotation, the average number of segment boundaries varies from 2.8 to 4.5 (average: 3.6), counting also the dialogue-final boundaries after the last topical segment of each dialogue. (Thus, these numbers correspond to the number of proposed topical segments.) The κ-agreement on topical boundaries is about 0.38 on average (see section 2.2.4).[20] We also note that the average length of the topical segments varies to a large extent, depending on the corpus: for CALLHOME and NEWSHOUR, the segments are rather short (300-500 words), for CROSSFIRE and GROUP MEETINGS, they are substantially longer (900-1400 words).

As a first starting point, we created a gold standard from these individual annotations automatically in the following way (using all the six annotators here, if available): For each boundary location, we include it into the gold standard if it is marked at least by half of the annotators which marked the particular dialogue. We set a flexible look-ahead window to avoid exclusion of close but not exact matches.[21] This yielded a total of 83 boundaries (or, equivalently, 83 topical segments.)

In a second step, the two annotators creating the human gold standard took this automatically created list of boundaries and slightly modified it to create the topic boundary human gold standard. They ignored 3 of the 83 boundaries, shifted 7 of them minimally (by at most 2 turns) and 1 to a larger extent. Thus, the human gold standard contains 80 topical boundaries, 57 of which are dialogue-internal and 23 are at the end of each dialogue. We decided to always include these end-boundaries here, also for the evaluations described below, since this is the only way to yield meaningful results for dialogues which contain only a single segment.[22] For all of our evaluations in this section, we use this human gold standard of topi-

[20] Dialogue-final boundaries were disregarded for this computation.

[21] The formula for this look-ahead l was: $l = 2 + \mathrm{int}(\frac{50}{\mathrm{avg\_words\_per\_turn}})$, yielding, e.g., 4 for dialogues with an average turn length of about 25 words and 7 for dialogues with an average turn length of about 10 words.

[22] If we were to exclude the trivial end-boundaries, the $recall$ measure would not be defined in these cases (due to a zero denominator).

cal boundaries as a reference.

3.5.4 Construction of stop word lists

Stop words have been successfully used in many IR systems, as well as topic segmenters and summarizers for written language. By eliminating high frequency closed class words (words with little or no referential meaning), there are two resulting advantages: (1) the system only focuses on (potentially) relevant information units (words), and (2) the vectors for words get considerably smaller, speeding up computation.

For the experiments, both in this section as well as for the evaluation of the summarization system (section 4.1), we used five different stop lists.

1. the original *Smart* list (Salton, 1971) (SMART-O)
2. a manually edited stop list based on *Smart* (Salton, 1971) (SMART-M)
3. a stop list with all closed class words from our POS lexicon (POS-O)
4. a manually edited stop list based on our POS lexicon and frequent words in the CALLHOME training corpus (POS-M)
5. an empty stop list

We scanned the POS lexicon we use for the POS tagger for all word entries which have *at least* one tag which does *not* belong to the 4 main open word classes of nouns (N*), verbs (V*), adjectives (J*), or adverbs (R*). Out of a total of 12206 entries in the lexicon, 398 (3.26%) fall in this category. For the remainder of this section, we call these words the "closed class words", even though many of those may not be considered as such by linguists; they (also) might be ambiguous between open class and closed class, an example being "thank" as part of the phrase "thank you" which could be considered a closed class "conventional phrase".

The original *Smart* stop list (file: `common_words`) contains 571 words (SMART-O). For the manually edited list (SMART-M), we first scanned the list to remove words which obviously should not be stop words for our purposes, including single letters (except "I" and "a"): these can be part of abbreviations such as "I R S" or "A B C" (which are usually transcribed as individual letter words), numbers, as well as some open class words which clearly can convey meaning in some contexts, such as, e.g., "allow". These decisions are certainly subjective and not always easy to make but we tried to be as conservative as possible in this removal phase to keep the list as close as possible to the original one. In total, 109 open class words and 40 closed class words were removed.[23] Second, we added a list of 8 short forms starting with an apostrophe ("'ll", "'d"...) plus "n't" which were not part of the list in the first place but are common tokens in the spontaneous dialogues. Further, 1 duplicate word was removed ("would"). The new edited list thus contains $571 - 109 - 40 + 8 + 1 - 1 = 430$ words (SMART-M).

Another stop word list was based on the list of 398 closed class words from the POS lexicon (POS-O). For the manually edited version of this list, we first added the 464 open class words from the 653 most frequent words from the CALLHOME training corpus.[24] Since the short forms starting with an apostrophe were already in this combined list, none had to be added. We went through the same removal phase as for the SMART stop list, but were much more rigorous here since the initial list had a large number of clearly content indicating words. Of course, we removed all the words which had already been removed from the initial SMART list before (91 words). A total number of 286 open class words and 27 closed class words were removed manually, thus the final stop list comprised $398 + 464 - 91 - 286 - 27 = 458$ entries.

There were 190 words in the SMART-M list not contained in the POS-M list, 218 words in the POS-M list not contained in SMART-M, and 240 words contained in both lists. Thus, less than half of the words in these two lists are actually different.

3.5.5 Evaluation

In the evaluation of the topic segmentation module, we used a *tolerance window* of 70 words to either side of the true (gold standard) boundaries where hypothesized boundaries were also considered to be correct. As metrics we used the standard definitions of precision (P) and recall (R) (B_c=correctly identified boundaries,

[23]The original list has 354 open class and 217 closed class words, a 62:38 split.

[24]Total tokens in 80 dialogues: approx. 180,000; occurrence threshold=20, token coverage=158k (88%), closed class words=189, open class words=464.

B_i=all identified boundaries, B_m=all marked boundaries (by human annotators)):

$$P = \frac{B_c}{B_i} \qquad R = \frac{B_c}{B_m} \tag{3.2}$$

As a single metric, the F_1-score was used. Note that for reporting results on multiple dialogues, we use the *micro-average* method of scoring, i.e., adding up B_c, B_i, B_m first and then computing precision, recall, and F_1-score, rather than just computing the arithmetic means of the individual dialogues' scores (*macro-average*).

The following parameters have to be tuned to get optimal performance for the TextTiling algorithm:

- stop word list: which of the five stop word lists to use
- blocksize: how many tokens to incorporate in each block of text (to the right and left of a potential boundary)
- sample distance: distance (in words) between two score computations: this speeds up the computation and also serves as a smoothing parameter
- α (in $\bar{s}-\alpha\sigma$): a higher α generates more boundaries and hence increases recall (at the expense of precision)
- smooth width: how many scores to include in the low-pass filter smoothing (to the right and to the left of the current score)
- smooth cycles: how often to iterate the smoothing operation

In order to optimize these parameters for the TextTiling algorithm, we ran a leave-one-out crossvalidation experiment for each of the 5 sub-corpora: in each of the k runs, one of the dialogues serves as test set and the remaining $k-1$ dialogues as training set. Parameters were optimized on the training sets (seen data) and then applied to the test sets (considered as unseen). Further, we determined for each of the 5 sub-corpora, which set of parameters yielded the best results on the majority of the cross-validation runs (see Table 3.20). We then computed the performance for the sub-corpora using this set of parameters. Also, we compare these scores against a *random* and an *equal-spaced* baseline. The random baseline means that boundaries were inserted randomly in the dialogues (but using the average number of topical segments as a rough target), equal means that boundaries were inserted uniformly in the dialogues, with equal distance from each other (again, using information about the average number of topical segments). A third baseline was constructed which we call the *equal-random* baseline: While boundaries are inserted randomly here, too, this method tries to establish a distance between boundaries which reflects the average topical segment length (of the respective sub-corpus). It can be considered also as an equal baseline with "noisy perturbations" of the actual boundary location. As can be seen from Table 3.19, this *equal-random* baseline is almost always the best baseline.

As for the results of the cross-validation, Table 3.19 shows that for CALLHOME, topic segmentation works very well: there is a strong improvement of the scores over all three baselines. However, for NHOUR, XFIRE, and G-MTG, the *equal*-baseline is hard to outperform for this algorithm.

We conducted a small qualitative analysis of some English CALLHOME dialogues to determine some sources of errors in the algorithm. We made the following observations:

- some words occur *across* segment boundaries, e.g., because the dialogue partners *use* them as "anchors" to shift the topic
- some segments are just too short (or too long, in some cases) to be able to be detected by a fixed window size algorithm
- some problems are caused by synonyms (e.g. kids vs. children) or shortcomings in the stemming procedure (e.g., years vs. year)
- there are potential features which could be exploited in an algorithm which combines them with the text similarity (e.g.: number of new words introduced per turn (or sequence of n turns), lexical chains, cue words/phrases, cue speech acts)
- sometimes, potential boundaries are "smoothed out" (although this is done in favor of the *overall* performance improvement of the algorithm)
- the algorithm tends to detect subtopics if they are fairly sizeable; this is certainly a matter of judgement and it is likely to vary between different human annotators whether boundaries are "main" or "sub"-topical

	8E-CH	4E-CH	NHour	XFire	G-Mtg
dialogues	8	4	3	4	4
number of topics	28	23	8	14	7
average topic length	482	331	459	904	1346
standard deviation of topic length	238	152	236	637	694
random baseline (F_1)	0.43	0.41	0.50	0.37	0.60
equal-random baseline (F_1)	0.47	0.49	0.53	0.39	0.53
equal baseline (F_1)	0.43	0.42	0.35	0.33	0.44
seen data (train set avg.) (F_1)	0.76	0.89	0.88	0.61	0.80
unseen data (test set avg.) (F_1)	0.59	0.76	0.53	0.40	0.47
best parameters (F_1)	0.75	0.86	0.59	0.58	0.67

Table 3.19
Topic segmentation results, using human gold standard boundaries as reference.

	8E-CH	4E-CH	NHour	XFire	G-Mtg
blocksize	300	150	150	150	300
sample distance	20	30	30	90	50
smooth iterations	2	2	2	2	2
smooth width	2	1	1	2	2
α	1	1	.5	.5	.5
stop word list	POS-m	POS-o	POS-o	empty	Smart-o

Table 3.20
Best overall topic segmentation parameters for each of the 5 sub-corpora.

3.6 Sentence Ranking and Selection

This component's purpose is to determine weights for terms and sentences, to rank the sentences according to their relevance within each topical segment of the dialogue, and finally to select the sentences for the summary output according to their rank, as well as to other criteria, such as question-answer linkages, established by previous components. This module performs the summarization task proper, in that it identifies the most salient parts of any given dialogue segment. We use this component in isolation (and without using any information from other system components) as one of the baselines for our global system evaluation.

3.6.1 Tokenization

Additionally to the tokenization rules for the global system (section 3.2), we apply a simple 6 character truncation for stemming and use a stop word list to eliminate frequent non-content words. In the experiments, we used the five stop word lists mentioned before (section 3.5.4): SMART-O, SMART-M, POS-O, POS-M, and EMPTY.

3.6.2 Term and sentence weighting

The basic idea of determining the most relevant sentences within a topical segment is as follows: First, we compute a vector of word weights for the segment tf_q (including all stemmed non stop words) and do the same for each sentence (tf_t), then we compute the similarity between sentence and segment vectors for each sentence. That way, sentences that have many words in common with the segment vector are rewarded and receive a higher relevance weight.

To minimize redundancies, we use a version of the *maximum marginal relevance* (MMR) algorithm (Carbonell, Geng, and Goldstein, 1997; Carbonell and Goldstein, 1998), where emphasis is given to sentences which contain many highly weighted terms for the current segment (salience) and are sufficiently dissimilar to previously ranked sentences (diversity or anti-redundancy). The MMR formula is given in Equation 3.3. It describes an iterative algorithm and states that the next sentence to be put in the ranked list will be taken from the sentences which were not yet ranked (t_{nr}) and has the following properties: it is (a) maximally similar to a query, and (b) maximally dissimilar to the sentences which were already ranked (t_r). We use the topical segment word vector $\mathrm{tf_q}$ as query vector. The λ-parameter ($0.0 \leq \lambda \leq 1.0$) is used to trade off the influence of salience vs. redundancy.

Both similarity metrics (sim_1, sim_2) are inner vector products of (stemmed) term frequencies (see equations 3.4 to 3.8); $\vec{\mathrm{tf}}_\mathrm{t}$ is a vector of stem frequencies in a sentence; f_s are the in-segment frequencies of a stem; f_{smax} are maximal segment frequencies of any stem in the segment. sim_1 can be normalized in different ways: (a) to yield a cosine vector product (division by product of vector lengths), (b) division by number of content words,[25] and (c) no normalization. The three formulae for $\mathrm{tf_s}$ (FREQ, SMAX, and LOG) are inspired from Cornell University's SMART system (Salton, 1971).

Further, inverse document frequency (idf) values with respect to a collection of topical segments — either the current dialogue or a set of dialogues — can optionally be multiplied onto the $\mathrm{tf_s}$-vectors; N_{seg} is the total number of topical segments in the idf-corpus, i_{seg} is the number of segments where the token i appears at least once. The effect of using idf-values is to boost those words which are (relatively) unique to any given segment over those which are more evenly distributed across the corpus.

$$nextsentence = \arg\max_{t_{nr,j}}(\lambda sim_1(query, t_{nr,j}) - (1-\lambda)\max_{t_{r,k}} sim_2(t_{nr,j}, t_{r,k})) \quad (3.3)$$

$$sim_1 = \frac{\vec{\mathrm{tf}}_\mathrm{q}\vec{\mathrm{tf}}_\mathrm{t}}{\left|\vec{\mathrm{tf}}_\mathrm{q}\right|\left|\vec{\mathrm{tf}}_\mathrm{t}\right|} \quad \mathrm{or} \quad \frac{\vec{\mathrm{tf}}_\mathrm{q}\vec{\mathrm{tf}}_\mathrm{t}}{1+\sum_\mathrm{i}\mathrm{tf}_\mathrm{i,t}} \quad \mathrm{or} \quad \vec{\mathrm{tf}}_\mathrm{q}\vec{\mathrm{tf}}_\mathrm{t} \quad (3.4)$$

$$sim_2 = \frac{\vec{\mathrm{tf}}_\mathrm{t1}\vec{\mathrm{tf}}_\mathrm{t2}}{\left|\vec{\mathrm{tf}}_\mathrm{t1}\right|\left|\vec{\mathrm{tf}}_\mathrm{t2}\right|} \quad (3.5)$$

$$\vec{\mathrm{tf}}_\mathrm{q} = \vec{\mathrm{tf}}_\mathrm{s}\vec{\mathrm{idf}}_\mathrm{s} \quad (3.6)$$

$$\mathrm{tf_{i,s}} = \mathrm{f_{i,s}} \quad \mathrm{or} \quad 0.5 + 0.5\frac{\mathrm{f_{i,s}}}{\mathrm{f_{smax}}} \quad \mathrm{or} \quad 1 + \log \mathrm{f_{i,s}} \quad (3.7)$$

$$\mathrm{idf_{i,s}} = 1 + \log\frac{N_{seg}}{i_{seg}} \quad \mathrm{or} \quad \frac{\mathrm{N_{seg}}}{\mathrm{i_{seg}}} \quad (3.8)$$

[25]To avoid division by zero, we add 1 to every sentence length.

Emphasis factors

Every sentence's similarity weight (sim_1) can be (de-)emphasized, based on a number of its properties. We implemented (optional) emphasis factors for

- leading $N\%$ of a segment's sentences;
- detected questions and answers;
- detected false starts; and
- individual speakers.

These parameters can serve to fine tune the system for particular applications or user preferences. E.g., if the false starts are deemphasized, they are less likely to trigger a question being linked to them in the linking process. If questions and answers are emphasized, more of them will show up in the summary, increasing its coherence and readability. In a situation, where a particular speaker's statements are of higher interest, his sentences can be emphasized, as well.

To keep the sim_1-scores in the interval [0,1] (which is necessary since sim_2, being a cosine vector product, always is in [0,1], and the effect of λ could else not be seen), we divide them by the maximum of all sim_1-scores in a segment after initial computation and application of the various emphasis factors described here.

3.6.3 Q-A linking

While generating the output summary from the MMR-ranked list of sentences, whenever a question or an answer is encountered (detected before by the Q-A detection module), the corresponding answer/question is linked to it and moved up the relevance ranking list to immediately follow the current question/answer. If any sentence of the current linkage is part of another Q-A linkage, the linkages are repeated until no further questions or answers can be added to the current linkage cluster.

3.6.4 Summary types

DIASUMM can generate three different types of summaries, the two main forms being (i) the CLEAN summary which is based on the automatically segmented sentences with all Q-A linkings performed and all disfluencies being removed, and (ii) the TRANS summary which is entirely based on the original transcript (no disfluency removal, sentence boundaries correspond to original turn boundaries, no Q-A linking).

The third version (iii) is a phrasal summary in telegraphic style (NPTELE), which renders the sentences in the same ranking order as the CLEAN summary, but which reduces the output to some extent: depending on the corpus, only certain kinds of phrases (determined by the chunk parser) are rendered in the summary output. This corresponds to an orthogonal text reduction *within* a sentence, as opposed to selecting sentences from a topical segment which is the task of the information condensation (MMR) module. For the two more formal corpora (NEWSHOUR and CROSSFIRE), we include noun phrases with a minimum length of two tokens and prepositional phrases; for the more informal corpora (CALLHOME and GROUP MEETINGS), we additionally include shorter NPs (to cover the more frequent and more relevant personal pronouns) and VPs, as well.[26] This reflects our analysis of differences between these two sets of corpora which we discussed in section 2.1: the two more formal sub-corpora are more "nominal", the other sub-corpora more "verbal" and "pronominal" in style.

As a second baseline for evaluation, complementing the MMR baseline, we also use a simple LEAD summary: Here, the summary just consists of the first N words of a given segment.

In Figure 3.3 we show examples of a LEAD, TRANS, CLEAN and NPTELE (phrasal) summary, generated from a CALLHOME dialogue. All these summaries have a length of 13.8% of the original transcript (34 of 246 words).

[26]The term VP here means something like "verbal cluster", "verbal chunk", and does *not* correspond to the notion of a verb phrase. Example of a verbal chunk: ...[didn't really like]... .

```
LEAD:
 39 b :   yeah well now get this we might go to live in switzerland
 40 a :   oh really
 41 b :   yeah because they've made him a job offer there and at
          first he's thinking nah he wasn't [...]

TRANS:
 40 b :   Yeah because they've made him a job offer
          there and at first he's thinking nah he
          wasn't going to take it but now he's like
 44 b :   And then you know the [...]

CLEAN:
 56 b :   Now get this we might go to live in switzerland
 59 b :   They've made him a job offer there
 60 b :   At first he's thinking
 63 b :   Maybe he could get [...]
 65 b :   The swiss phone company whatever and telefonika

PHRASAL:
 56 b :   now get we might go to live in switzerland ...
 59 b :   they've made him a job offer ...
 60 b :   he's thinking ...
 63 b :   he could get in his foot in the door ...  they [...]
 65 b :   the swiss phone company ...  telefonika ...
```

Figure 3.3

Example summaries of 13.8% length: LEAD, TRANS, CLEAN and NPTELE. Notes: The turn-IDs are just indicating the relative position of the turns within the original text and do not always correspond to the turn numbers of the original or to the turn numbers of the other summaries. The '[...]' marks the position in those sentences where the length threshold for a summary was reached.

3.6.5 System tuning

This section describes how we arrive at an optimal parameter setting for each sub-corpus (CALLHOME, NEWSHOUR, CROSSFIRE, GROUP MEETINGS). We want to establish a MMR-baseline for the global system evaluations with which we can then compare the results of the entire DIASUMM system. Note that for all the tuning experiments described in this section, we did not make use of any other DIASUMM component, namely the disfluency detection and removal, sentence boundary detection, question-answer linking, and topic segmentation. All experiments were based on the human gold standard with respect to topical segments. We only used the `devtest` set for the 4 sub-corpora here: 8E-CH, DT-NH, DT-XF, and DT-MTG.

Since the length of turns varies widely, one could argue that performance might increase by splitting overly long turns evenly into shorter chunks. (Note that we don't do sentence segmentation for this baseline.) This has been done by Valenza et al. (1999) who experimented with lengths of 10-30 words per extract fragment. We add this option as an additional parameter to the system. If the parameter is set to N words, turns with a length $l >= 1.5N$ get cut into pieces of lengths N iteratively until the last remaining piece is $l < 1.5N$.

Evaluation metric

All evaluations are based on topically coherent segments from the dialogue excerpts of our corpus. As mentioned before, the segment boundaries were chosen from the human gold standard, for the purpose of the global system evaluation.

For each segment s, for each annotator a, we define a boolean word vector of annotations $w_{s,a}$, each component $w_{s,a,i}$ being 1 if the word w_i is part of a nucleus-IU or a satellite-IU for that annotator and segment, and 0 otherwise.

We then sum over all annotators' annotation vectors and normalize them by the number of annotators per segment (A) to obtain the average relevance vector for segment s, r_s:

$$r_{s,i} = \frac{\sum_{a=1}^{A} w_{s,a,i}}{A} \tag{3.9}$$

To obtain the summary accuracy score $sa_{s,N}$ for any segment summary with length N (automatically generated or produced by a human annotator), we multiply the boolean summary vector $\mathrm{summ_s}$ with the average relevance vector r_s, and then divide this product by the sum of the N highest scores within r_s (maximum achievable score), $\mathrm{rsort_s}$ being the vector r_s sorted by relevance weight in descending order:

$$sa_{s,N} = \frac{\mathrm{summ_s r_s}}{\sum_{i=1}^{N} \mathrm{rsort_{s,i}}} \tag{3.10}$$

It is easy to see that the summary accuracy score always is in the interval $[0.0, 1.0]$.

Summary accuracy scores are then averaged over all topical segment summaries of a sub-corpus.

Phase 1

In the first phase, we optimized the term weighting parameters, while holding the summary size constant to 15% and varying the MMR-λ only slightly (0.9 vs. 1.0).

The list of parameter settings for these experiments is given here:

1. term weight type: freq, smax, log
2. normalization: cos, length, none
3. idf type: corpus, dialogue, none
4. idf method: log, mult
5. MMR-λ: 0.9, 1.0
6. extract span: 10, 20, 30, original turn

Phase 2

Here, we start with the optimized parameters from Phase 1 and tune for MMR-λ and the five different stop word lists mentioned before.

The list of parameter settings for this phase is:

Table 3.21
Optimally tuned parameters for MMR baseline system (tuning on `devtest` set sub-corpora).

	8E-CH	DT-NH	DT-XF	DT-MTG
term weight type	smax	smax	smax	smax
normalization	cos	no	cos	no
idf type	corpus	corpus	corpus	corpus
idf method	log	log	mult	log
extract span	20	orig	25	orig
MMR-λ	0.85	0.8	1.0	0.8
stop list	SMART-M	POS-M	POS-M	POS-M
lead emphasis	1.0	1.0	1.0	2.0

1. MMR-λ: 0.8, 0.85, 0.9, 0.95, 1.0
2. stop lists: SMART-O, SMART-M, POS-O, POS-M, EMPTY

Phase 3

The third and last phase in our baseline tuning involves the LEAD-emphasis-parameter. We used the optimized parameters from Phase 2 and varied the lead factor from 1.0 to 5.0, keeping a constant lead-length of 20%. Again, only summaries of size=15% were evaluated.

Table 3.21 shows the parameter settings which were determined to be optimal for the MMR baseline system (TRANS summaries).

3.7 System Integration and Performance

The majority of the system components are implemented in `Perl5`, except for the C4.5 decision tree (Quinlan, 1992), the POS based chunk parsers (Ward, 1991; Gavaldà, 2000), and the POS tagger (Brill, 1994), which were implemented in C/C++ by the respective authors.

The DIASUMM system was developed and evaluated on Sun/Solaris platforms, but was also integrated into the *Meeting Browser* (Waibel, Bett, and Finke, 1998; Waibel et al., 2001) which runs on PC/Windows platforms. The Meeting Browser is a graphical user interface which enables the recording and automatic transcription of speech input (via the JANUS recognition toolkit), as well as the archiving, indexing, searching, and displaying of such recordings. The DIASUMM system runs as a server and adds a flexible summarization feature to the Meeting Browser: users can interactively drill-down on the contents of meetings (and other recordings), they can select from different versions of summaries (basic=MMR summary, clean=DIASUMM summary, noun phrase summary, topic based keyword summary[27]), they can focus the summaries on self-defined keywords (query-specific summaries), and listen to the corresponding audio portions of displayed summaries. Further features are optional additions of stop words to the stop word list, varying the Q-A emphasis, changing summary size both globally and locally (relative to a topic), and using ASR confidence scores (if available) to improve the summary accuracy and reduce its word error rate.

To establish the overall system performance, as well as the relative time contribution of the different modules, we measured the system runtime on a 300 MHz Sun Ultra60 dual processor workstation with 1 GB main memory, summarizing all 23 dialogue excerpts from our corpus. We ran two kinds of evaluations: (a) DIASUMM in evaluation-mode, i.e., the entire script with all its components, including the automatic performance evaluation module following DIASUMM, but without the topic segmentation module (since this is not relevant in evaluation-mode); (b) in standard run mode, where we do use the topic segmentation module, but do not run the post-summarization evaluation script. Furthermore, we also ran the script on a PentiumIII 900MHz PC (Windows 2000 Professional), in standard run mode only. Table 3.22 presents the results of these evaluations. The average runtime for the whole system was about 16 seconds on the Sun and about 6 seconds on the PC (about 1% of real speaking time since the average length of a dialogue is about 10 minutes). The longest time is consumed by the information condensation component (MMR) (33–38%), followed by the sentence boundary detection component (19–23%), the false start detection component (13-23%), and the POS tagger (13-17%). The relative contribution of the other components is typically less than 5% each of the total run time.

[27] A list of the most salient keywords within a topic.

	Sun/eval	Sun/standard	PC/standard
CPU Clock (MHz)	300	300	900
Total time (sec)	16.8 (100.0)	15.7 (100.0)	6.4 (100.0)
POS Tagger	2.4 (14.5)	2.7 (16.9)	0.9 (13.6)
Sentence Boundary Detection	3.5 (20.7)	3.5 (22.4)	1.3 (19.7)
False Start Detection	2.2 (13.2)	2.3 (14.6)	1.5 (23.1)
Q-A Detection	0.7 (3.9)	0.7 (4.1)	0.2 (2.7)
Repetition Filter	0.7 (4.2)	0.5 (3.0)	0.2 (2.7)
Topical Segmentation	—	0.7 (4.1)	0.3 (4.8)
Information Condensation (MMR)	6.1 (37.6)	5.5 (34.8)	2.1 (33.3)
Performance Evaluation	1.0 (6.0)	—	—

Table 3.22
Average DIASUMM run times in seconds (in brackets: relative run time in percent).

Not until we suppress the question "why",
we often become aware of the important *Tatsachen;*
which then lead to an answer in our investigations.
L.Wittgenstein

Chapter 4

Evaluations

Traditionally, the evaluation of summarization systems has been performed in two major ways: (a) intrinsically, measuring the amount of the core information preserved from the original text (Kupiec, Pedersen, and Chen, 1995; Teufel and Moens, 1997); and (b) extrinsically, measuring how much the summary can benefit in accomplishing another task, e.g., finding a document relevant to a query or classifying a document into a topical category (Mani et al., 1998).

In this thesis, we focus on intrinsic evaluation exclusively. That is, we want to assess, how well the summaries preserve the essential information contained in the original text. As other studies have shown (Rath, Resnick, and Savage, 1961; Marcu, 1999), the agreement between human annotators about which passages to choose to form a good summary is usually quite low. Our own findings, reported in section 2.2.4, support this in that the inter-coder agreement, here measured on a word level, is quite low, as well.

We decided to minimize the bias that would result from selecting either a particular human annotator, or even the manually created gold standard as a reference for automatic evaluation, but instead weigh all annotations from all human coders equally. Intuitively, we want to reward summaries which contain a high amount of words considered to be relevant by the largest number of annotators (see section 3.6.5).

This chapter consists of the following sections which evaluate different dimensions of our system:

1. a global comparison of DIASUMM against a human-defined gold standard and two different baselines (LEAD, MMR) (section 4.1)

2. an evaluation of the influence of imperfect topical boundaries on summary accuracy (section 4.2)

3. an investigation on summary word error rate reduction, using speech recognizer confidence scores (section 4.3)

4. an evaluation about the influence of cross-speaker information linking on summary informativeness and summary coherence (section 4.4)

5. a user study determining answer time and answer accuracy on multiple choice questions related to core information in dialogue segments (section 4.5)

4.1 Global System Evaluation

While chapter 3 was concerned with the design and evaluation of the individual system components, the goal here is to evaluate and analyze the quality of the global system, with all its components combined.

In this section, we compare the full DIASUMM system with the MMR baseline system, which operates without any dialogue specific components, and a LEAD baseline, which just includes the first N words of a given topical segment into the summary. (The MMR baseline system corresponds to only running the sentence selection and ranking component.)

We described the optimization and fine tuning of the MMR system in section 3.6.5. The second column of Table 4.1 presents the average relevance scores, averaged over the 5 summary sizes of 5, 10, 15, 20, and 25 percent, for the 4 `devtest`-set and the 4 `eval`-set sub-corpora. We also give, as a comparison, the LEAD score averages for the same sizes of summaries, where the summary contains the first N percent of the word tokens within a segment.

We used the optimized baseline MMR parameters and varied the emphasis parameters for (a) false starts, (b) lead factor, and (c) Q-A sentences, to optimize the CLEAN-summaries — the 'standard' DIASUMM summaries — further (using only the `devtest`-set for this optimization).

For each corpus in the `devtest`-set, we determined the optimal parameter setting and report the corresponding results also for the `eval`-set sub-corpora. Table 4.1 provides the comparison of the average scores for LEAD, baseline MMR,

Table 4.1
Average summary accuracy scores. `devtest`-set and `eval`-set sub-corpora on optimized parameters, comparing LEAD, MMR baseline, DIASUMM, NPTELE, and the human gold standard. Note: The gold standard summaries are based on all tokens from nucleus-IUs and have a fixed size.

sub-corpus	LEAD	MMR	DIASUMM	NPTELE	gold (size in %)
8E-CH	0.463	0.545	**0.597**	0.575	0.709 (13.1)
DT-NH	0.386	**0.637**	0.554	0.511	0.791 (20.9)
DT-XF	0.516	**0.595**	0.541	0.521	0.764 (11.4)
DT-MTG	0.488	0.594	**0.606**	0.588	0.705 (14.9)
4E-CH	0.438	0.526	**0.614**	0.602	0.793 (12.9)
EVAL-NH	**0.692**	0.526	0.506	0.536	0.850 (14.4)
EVAL-XF	0.378	0.564	**0.566**	0.527	0.790 (13.9)
EVAL-MTG	0.324	0.449	**0.583**	0.545	0.704 (16.0)

Table 4.2
Best emphasis parameters for the DIASUMM system, trained on the `devtest`-set.

corpus	false start	Q-A	lead factor
CALLHOME	0.5	1.0	2.0
NEWSHOUR	0.5	2.0	1.0
CROSSFIRE	0.5	1.0	1.0
GROUP MEETINGS	0.5	1.0	3.0

DIASUMM system, NPTELE summaries, and the human gold standard (nucleus-IUs only, fixed length summaries). Table 4.2 shows the best parameter combinations for the DIASUMM summaries used in these evaluations.

We determined the statistical differences between the DIASUMM system and the two baselines (LEAD, MMR) for the `eval`-set, using the Wilcoxon rank sum test for each of the 4 sub-corpora. Comparisons were made for each of the five summary sizes within each topical segment. For the CALLHOME and GROUP MEETINGS sub-corpora, our DIASUMM system is significantly better than the MMR baseline ($p < 0.01$); for the two more formal sub-corpora, NEWSHOUR and CROSSFIRE, the difference is not significant. Except for the NEWSHOUR sub-corpus, both the MMR baseline and the DIASUMM component perform significantly better than the LEAD baseline.

Figures 4.1, 4.2, 4.3, and 4.4 show the average performance of the following six system configurations, averaged over all topical segments and all summary

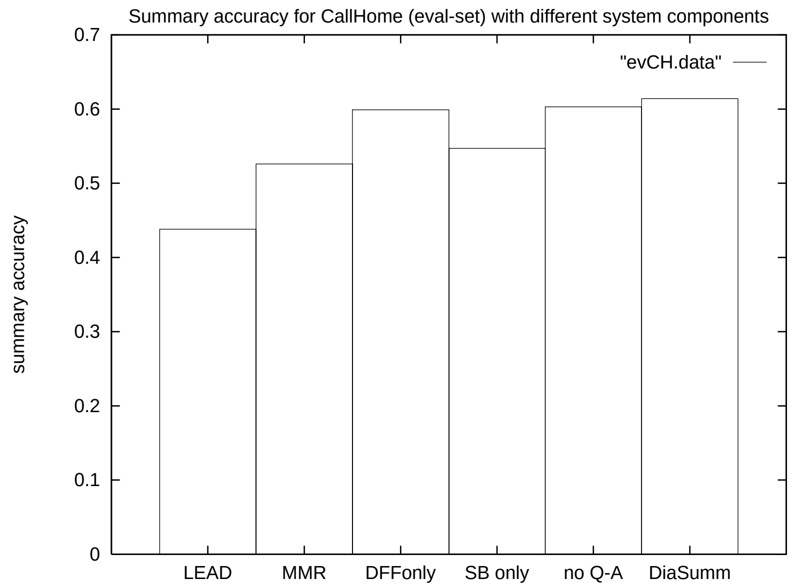

Figure 4.1
Average summary accuracy scores for different system configurations for the 4E-CH sub-corpus.

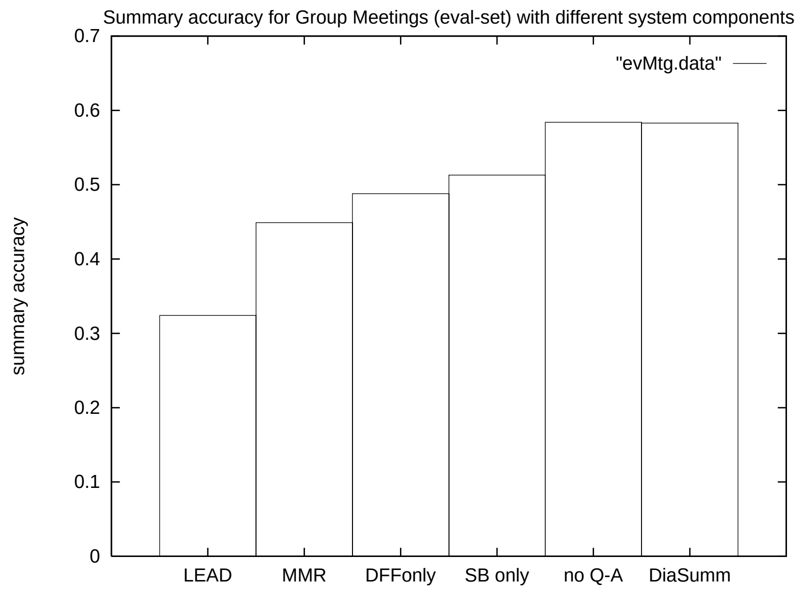

Figure 4.2
Average summary accuracy scores for different system configurations for the GROUP MEETINGS sub-corpus.

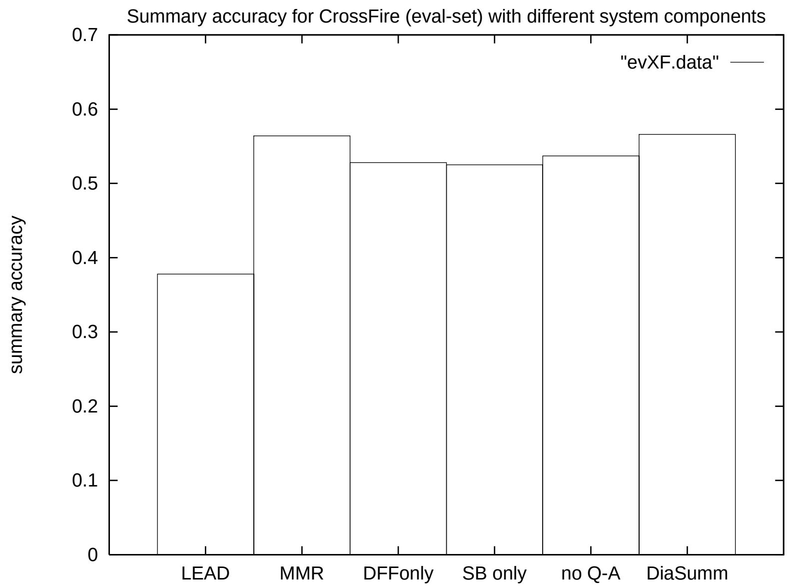

Figure 4.3
Average summary accuracy scores for different system configurations for the CROSSFIRE sub-corpus.

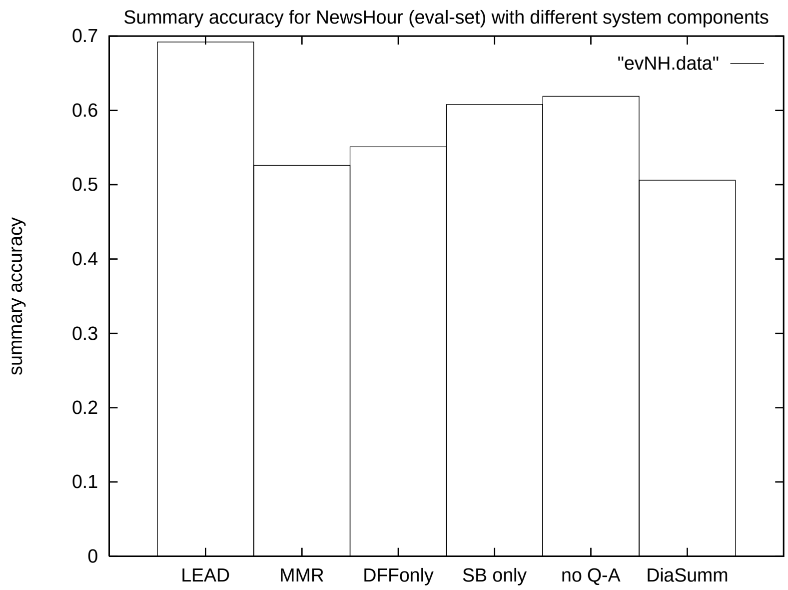

Figure 4.4
Average summary accuracy scores for different different system configurations for the NEWSHOUR sub-corpus.

sizes (5-25% length summaries; configurations 3-5: components used there are in *addition* to the core MMR summarizer):

1. LEAD: using the first N% of the words in a segment
2. MMR: the MMR baseline (tuned, see above)
3. DFF-ONLY: using the disfluency detection components (POS tagger, false start detection, repetition detection), but no sentence boundary detection and question-answer linking
4. SB-ONLY: using the sentence boundary module, but no other dialogue specific modules
5. NO-QA: a combination of DFF-ONLY and SB-ONLY: all preprocessing components used except for question-answer linking
6. DIASUMM: complete system with all components (all disfluency detection components, sentence boundary detection, and Q-A linking)

When we look at these graphs, we observe that in all sub-corpora, except for CROSSFIRE, the addition of either the disfluency components or the sentence boundary component improved the summary accuracy over the MMR baseline. [The reason this seems not to happen for CROSSFIRE is that the MMR baseline uses a limit on turn length (maximal extract span of 25 words, see tuning section above) — this yields to a much higher summary accuracy than in the case where the original turns would be used unchanged: Here, we obtain a summary accuracy of 0.519, which is slightly below the DFF-ONLY and SB-ONLY configurations.] As we would expect, given the much higher frequency of disfluencies in the two informal sub-corpora (CALLHOME, GROUP MEETINGS), the relative performance increase of DFF-ONLY over the MMR baseline is much higher here (about 10-15%) than for the two more formal sub-corpora (5% and below). Looking at the performance increase of SB-ONLY, we find marked improvements over the MMR baseline for those two sub-corpora which use the true original turn boundaries in the MMR baseline: GROUP MEETINGS and NEWSHOUR (>10%); for the two other sub-corpora, the respective improvement is rather small (<5%).

Furthermore, the combination of the disfluency detection and sentence boundary detection components (NO-QA) improves the results over the configurations DFF-ONLY and SB-ONLY.

The situation is much less uniform, when we add the question-answer detection component (this then corresponds to the full DIASUMM system): In the CROSSFIRE corpus, we have the largest performance increase (we also have the highest relative frequency of question speech acts here). For the two informal corpora, the change is only minor, for NEWSHOUR, the performance decreases substantially. We will see in section 4.4, however, that in general, for dialogues with relatively frequent Q-A exchanges, the summary accuracy (informativeness) does not change significantly when applying the Q-A detection component. On the other hand, the (local) coherence of the summary does increase significantly — but we cannot measure this with the evaluation criterion of summary accuracy used here.

To conclude, we have shown that using the dialogue specific components, with the possible exception of the Q-A detection module, can help creating more accurate summaries for more informal, casual, spontaneous dialogues. When facing more formal conversations (which may even be partially scripted), containing relatively few disfluencies, either a simple LEAD method or a standard MMR summarizer will be much harder to improve upon.

We can see, however, two main avenues to proceed in future work which may be of benefit for both formal and informal dialogue summarization: (i) the use of additional prosodic information such as stress and pitch (e.g., for improved sentence boundary or question detection), and (ii) the preparation of training data in the style of the disfluency annotated SWITCHBOARD database to better adapt to a particular genre of conversations.

4.1.1 Comparison against oracle performance

For the purpose of the global system evaluation, we have established a tuned MMR baseline and then compared to this baseline the various versions of the DIASUMM system, based on the optimal MMR parameters[1].

In this subsection, we will look at another interesting question, namely, how well our system performs in comparison to an oracle, i.e., an optimal system with the following idealizations:

[1] Except for the emphasis factors which were (re-)trained particularly for DIASUMM.

- perfect disfluency detection
- perfect sentence boundary detection
- perfect Q-A linking

We can easily derive these idealized versions by using the human corpus annotations already made for these three dimensions. When we want to compute the oracle performance of DIASUMM in different configurations — we choose the same 4 configurations as above: DFF-ONLY, SB-ONLY, NO-QA, and DIASUMM —, we have to decide which set of MMR parameters to use. We could use the same parameters as for our main experiments above, but since they were not tuned on the idealized system, but on the MMR baseline, they are most likely sub-optimal. Thus, we decided to re-train the system with idealized input with respect to the three main aspects: speech disfluency detection, sentence boundary detection, and question-answer linking, using the `devtest` sub-corpora only for this parameter tuning.

Parameter optimization proceeded in two phases. Since in the MMR baseline tuning before, the optimal term weight type always was "smax" and the optimal IDF type always was "corpus", we kept these two parameters fixed to their respective values. Further, the extract span parameter can be ignored, since we do not create the MMR baseline summaries here. In Phase 1, we tuned the following four parameters, using 15% summaries only:

- normalization: cos, length, none
- IDF method: log, mult
- MMR-λ: 0.8, 0.85, 0.9, 0.95, 1.0
- stop lists: SMART-O, SMART-M, POS-O, POS-M, EMPTY

In Phase 2, we use the optimized parameters of Phase 1 and tuned two emphasis parameters (the false start emphasis is immaterial due to perfect false start detection and removal): lead factor, and Q-A emphasis factor (both from 1.0–5.0), again using summary sizes of 15%. Table 4.3 shows the parameter settings which were determined to be optimal for the oracle DIASUMM system; they only differ slightly from the tuned MMR parameters in the major evaluation described above.

Finally, we ran DIASUMM with its "real" components, as well as with oracle components, on these optimized parameters. The results of these experiments for the `eval`-set sub-corpora are shown in the comparative Figures 4.5, 4.6, 4.7, and 4.8 (averages over all topical segments and summary sizes of 5-25%).

We observe that for the informal corpora (CALLHOME, GROUP MEETINGS), the DFF-ONLY versions perform better than the SB-ONLY versions; the converse is true for the formal corpora: here, the sentence boundary detection component makes a larger improvement than the disfluency detection components. This is readily explainable by the difference in data sets: The formal corpora contain much fewer disfluencies but have typically longer turns with more sentences per turn. Further, the combination of DFF-ONLY and SB-ONLY always leads to a better performance of the real system than each of the individual components by themselves. Finally, adding the Q-A linking component leaves the results almost unchanged (as observed in the comparative evaluations discussed above), except for a marked drop in performance for the NEWSHOUR corpus. We note, however, that for NEWSHOUR, the simple LEAD baseline yields by far the best results of all system configurations.

As for the difference between the oracle and the real system, we observe that for CALLHOME and NEWSHOUR, the differences are only minor: our system performs basically as good as the oracle system. For CROSSFIRE and GROUP MEETINGS, however, there is a marked difference between oracle and real system. We quantify these differences for the full DIASUMM systems in Table 4.4 for all sub-corpora except for NEWSHOUR, where the LEAD baseline is optimal already and even the oracle system does not yield better results than the MMR baseline.

The lowest absolute and relative gain over the MMR baseline (relative here means: compared to the oracle system gain over the MMR baseline) is obtained for the CROSSFIRE sub-corpus (13.6%), a formal corpus where the individual components of DIASUMM have a lower performance than for informal corpora. For GROUP MEETINGS, we achieve more than half (56.3%) of the possible maximum gain, and for CALLHOME, the real system performance basically matches the oracle system performance (93.3%).

For the NEWSHOUR and the CALLHOME corpora, we also observe that for the SB-ONLY system configurations, the oracle performance is slightly below the real system performance. We believe that this is due to the fact that the oracle sentence

Table 4.3
Parameters tuned for the DIASUMM system using optimal (oracle) information for disfluencies, sentence boundaries, and QA-pairs (tuning on `devtest` set sub-corpora).

	8E-CH	DT-NH	DT-XF	DT-MTG
normalization	cos	len	cos	no
idf method	log	log	mult	log
MMR-λ	0.8	1.0	1.0	0.9
stop list	SMART-O	SMART-O	SMART-M	SMART-M
lead emphasis	2.0	1.0	1.0	2.0
Q-A emphasis	1.0	1.0	1.0	1.0

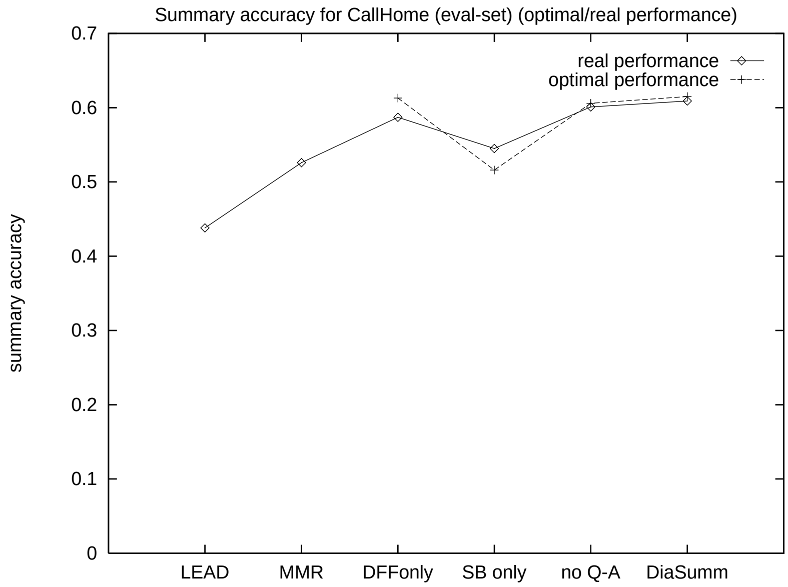

Figure 4.5
Average summary accuracy scores for different system configurations for the 4E-CH sub-corpus; comparing oracle performance and real system performance.

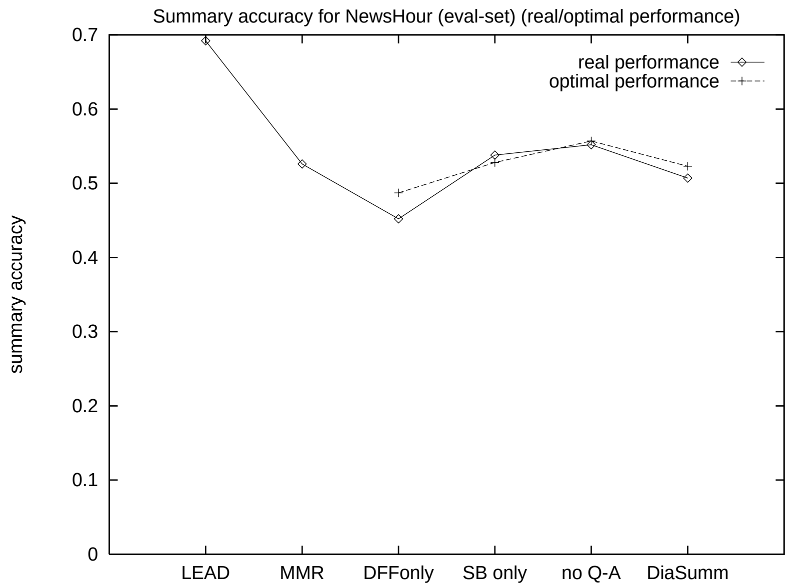

Figure 4.6
Average summary accuracy scores for different system configurations for the EVNHOUR sub-corpus; comparing oracle performance and real system performance.

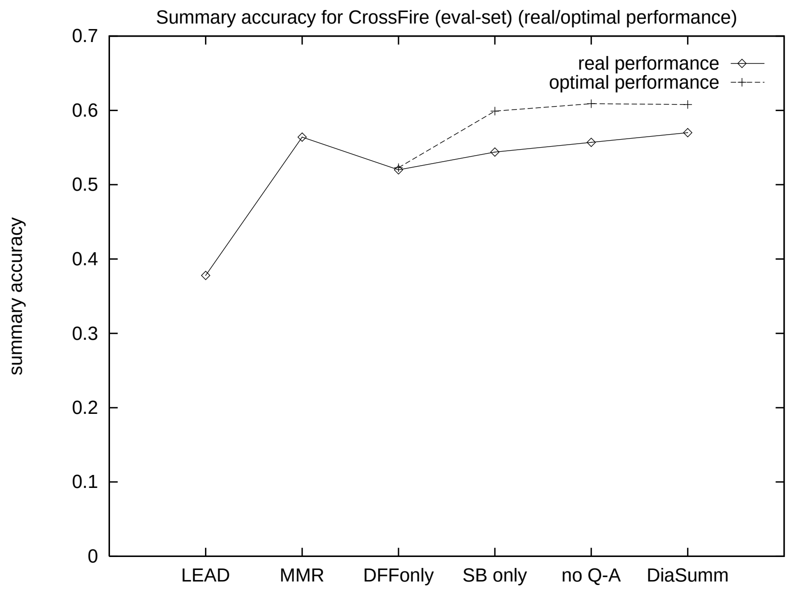

Figure 4.7
Average summary accuracy scores for different system configurations for the EVXFIRE sub-corpus; comparing oracle performance and real system performance.

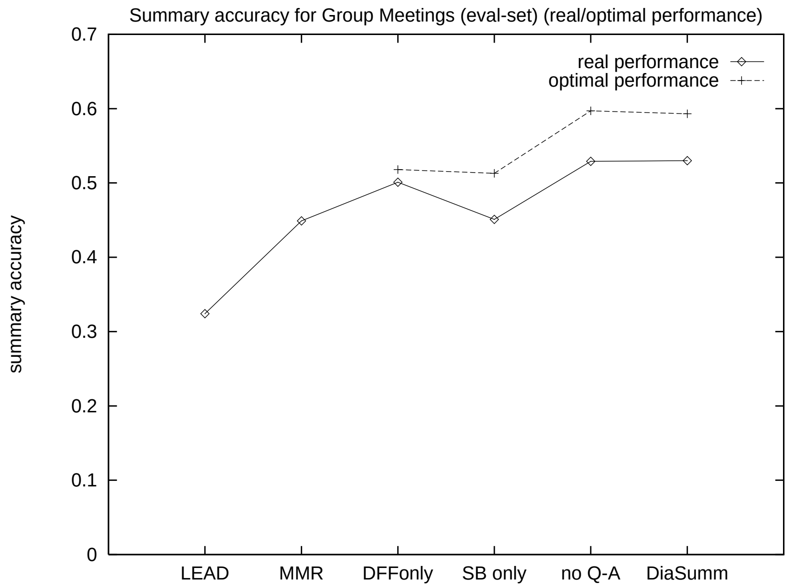

Figure 4.8
Average summary accuracy scores for different system configurations for the EVGMTG sub-corpus; comparing oracle performance and real system performance.

	MMR baseline	real DIASUMM	oracle DIASUMM	maximum gain	real gain	real gain (in %)
4E-CH	.526	.609	.615	.089	.083	93.3%
ev-XFire	.564	.570	.608	.044	.006	13.6%
ev-Mtg	.449	.530	.593	.144	.081	56.3%

Table 4.4
Relative performance improvement over the MMR baseline wrt. the oracle performance for 4E-CH, EV-XFIRE, EV-MTG.

Table 4.5
Average summary accuracy scores. `devtest`-set and `eval`-set sub-corpora, using automatic topic detection, comparing LEAD, MMR baseline, and DIASUMM (best scores in bold)

sub-corpus	LEAD	MMR	DIASUMM
8E-CH	0.349	0.463	**0.521**
DT-NH	0.525	**0.621**	0.422
DT-XF	0.526	**0.598**	0.527
DT-MTG	0.475	0.558	**0.631**
4E-CH	0.371	0.508	**0.550**
EVAL-NH	**0.691**	0.526	0.507
EVAL-XF	0.345	**0.519**	0.471
EVAL-MTG	0.324	0.432	**0.583**

segmentation is too aggressive with respect to the summary accuracy evaluation in that the sentence fragments produced are on average too small and that the larger extract spans generated by the real system are better suited for the purpose of summary generation. However, we think that not too much should be concluded from this, since these differences are too small to be statistically significant.

4.2 Influence of Imperfect Topical Boundaries

While all of the major evaluations in this thesis assume perfect topical boundaries (we use the boundaries from the human gold standard), in this section, we explore how much the system's performance would change when we use our automatic topic segmentation module, based on TextTiling, instead. We are interested in (a) performance change per se, but also (b) in performance differences between the LEAD, MMR and DiaSumm methods.

We ran the same evaluation as for the global evaluation above, but now used the topic segmentation module. We used the "best parameters" for this module, as described in section 3.5. The results of these experiments are shown in Table 4.5.

Table 4.6 shows the differences in percent, comparing automatic topic detection with ideal topic boundaries from the human gold standard. We see that there is a degradation of performance across the board (except for a rather high *gain* for dtNH-LEAD summaries), though not of a large magnitude. Also, we observe a

Table 4.6
Relative change of performance in percent when performing automatic topic detection.

sub-corpus	LEAD	MMR	DIASUMM
8E-CH	-24.62	-14.73	-12.73
DT-NH	36.01	-2.51	-23.83
DT-XF	1.94	0.50	-2.59
DT-MTG	-2.66	-6.16	4.13
4E-CH	-15.30	-6.10	-10.42
EVAL-NH	-0.14	0.00	0.20
EVAL-XF	-8.73	-7.98	-16.78
EVAL-MTG	0.00	0.00	0.00

relatively larger degradation for DiaSumm (compared to MMR). For the informal genres (CallHome, Meetings), DiaSumm still significantly outperforms the MMR baseline system.[2] For eval-XF, there is a trend ($p < 0.1$) towards the MMR baseline outperforming DiaSumm.

A second series of experiments concerned the question, how much of a gain the topic segmentation method yields compared to summarizing the whole dialogues as one single text each. While we did not optimize the DIASUMM parameters for this case, the results should at least give an indication of how important it is to determine topical boundaries in multi-topical settings of that kind. We set the lead-factor enhancement to 1.0 to avoid a bias towards the beginning of the dialogue and set a threshold of 0.1 for MMR computation (minimum turn-query similarity), to speed up computation. We further varied the MMR-λ between 0.6 and 1.0 and picked the best performing λ ($\lambda = 0.7$). The results of these experiments are shown in Table 4.7.

In general we want to caution the reader not to over-interpret these results since they are all obtained from *global* summaries, pertaining to whole dialogues, and hence the number of samples is very low ($n \leq 2$ for all cases except for 4E-CH and 8E-CH), unlike for all the other evaluations in this thesis, which pertain to topical segments.

- CALLHOME: Here, topical segmentation always helps, except for MMR baseline summaries with automatic topical segmentation. Also, the top-performing

[2] eval-Mtg does not change here since the dialogue excerpts are mono-topical in the first place and the topic segmentation module keeps it that way.

Table 4.7
Global summary accuracy (based on whole dialogue excerpts), comparing (a) global MMR, (b) automatic topic segmentation, and (c) standard evaluation mode (human gold standard topical boundaries).

	MMR			DiaSumm		
sub-corpus	1-topic	auto-segm.	standard	1-topic	auto-segm.	standard
8E-CH	0.526	0.490	0.546	0.510	0.531	0.574
4E-CH	0.497	0.502	0.540	0.546	0.574	0.601
devtestNH	0.565	0.654	0.661	0.519	0.567	0.540
devtestXF	0.535	0.605	0.593	0.536	0.569	0.558
devtestMtg	0.639	0.605	0.592	0.608	0.686	0.664
evalNH	0.427	0.517	0.517	0.606	0.474	0.477
evalXF	0.448	0.521	0.509	0.508	0.473	0.489
evalMtg	0.495	0.432	0.432	0.633	0.583	0.583

method is always the one using the topic boundaries from the human gold standard. In accordance with our earlier finding in the segment based evaluations, in 5 of 6 cases, the full DIASUMM system outperforms the MMR baseline. Thus, this advantage is neither caused by the fact that we use topical segments in the first place, nor that these segments are assumed to be ideal in most of our evaluations.

- Other sub-corpora: The picture for the remainder of the corpus is rather mixed and not uniform; but again, we have to keep in mind that we have at most 2 different dialogues to consider in these cases and cannot reach any firm or statistically significant conclusions. Furthermore, the topic segmentation module works best for the CALLHOME sub-corpora and not as well for the other sub-corpora, as we have seen in section 3.5. For the MMR-baseline, the general trend of improved performance when summarizing topical segments individually seems to hold for XF and NH, but not for the G-Mtg sub-corpora: topical segmentation in the latter case seems to hurt performance. For the full DIASUMM system, topical segmentation seems to be beneficial in the devtest sub-corpora, and rather harmful in the evaluation sub-corpora.

4.3 Reducing Summary Word Error Rate Using Speech Recognizer Confidence Scores

4.3.1 Introduction

For the most part of this thesis, we abstract away from the issue of speech recognition errors and look at manually created transcriptions of the dialogues instead. In this section, however, we analyze the behavior and performance of the DIASUMM system, when the input is ASR (automatic speech recognition) output. It is clear that when incorrect textual information is fed into the DIASUMM system, there will also be errors in the output. One of the crucial questions, however, is how to minimize the word error rate (WER) without compromising on the summary accuracy.

Several research groups have developed interactive browsing tools, where audio (and possibly video) can be accessed together with various types of textual information (transcripts, summaries) via a graphical user interface (Waibel, Bett, and Finke, 1998; Valenza et al., 1999; Hirschberg et al., 1999). With these tools, the problem of misrecognitions is alleviated in the sense that the user can always easily listen to the audio recording corresponding to a passage in a textual summary. In some instances, however, this approach may not be feasible or too expensive to pursue, and a short, stand-alone textual representation of the spoken audio may be preferred or even required. This section addresses in particular this latter case and (a) explores means of making textual summaries less distorted (i.e., reducing their word error rate (WER)), and (b) assesses how the accuracy of the summaries changes when methods for word error rate reduction are applied.

In related work, (Valenza et al., 1999) report that they were able to reduce the word error rate in summaries (as opposed to full texts) by using speech recognizer confidence scores. They combined inverse frequency weights with confidence scores for each recognized word. Using summaries composed of one 30-gram[3] per minute speaking time (approximately 15% length of the full text since the typical speaking rate is about 200 words per minute), the WER dropped from 25% for the full text to 10% for these summaries. They also conducted a qualitative study where human subjects were given summaries of n-grams of different length

[3] A 30-gram is a contiguous segment containing 30 words.

and also summaries with speaker utterances as minimal units, either giving a high weight to the inverse frequency scores or to the confidence scores. The utterance summaries were considered best, followed closely by 30-gram summaries, both using high confidence score weights. This suggests that not only does the WER drop by extracting passages that are more likely to be correctly recognized but also do summaries seem to be better which are generated that way.

While the results of (Valenza et al., 1999) are *indicative* for their approach, we want to investigate the benefits of using speech recognizer confidence scores in more detail and particularly find out about the *trade-off* between WER and summarization accuracy when we vary the influence of the confidence scores. In (Zechner and Waibel, 2000b), a paper preceding the discussion and results of this section, we addressed this trade-off for the first time in a clear, numerically describable way.

4.3.2 Evaluation metrics

The challenge of devising a meaningful evaluation metric for the task of audio summarization is that it has to be applicable to both the reference (human transcript) and the hypothesis transcripts (automatic speech recognizer (ASR) transcripts). We want to be able to assess the quality of the summary with respect to the relevance markings of the human annotators, as well as to relate this *summary accuracy* to the word error rate present in the ASR transcripts.

The approach we take is to *align* the words in the summary with the words in the reference transcript (w_a). Alignment for reference summaries is trivial since the words in the summary are a proper (and known) subset of the words in the original. For ASR transcript summaries, we have to determine the number of insertions or deletions between each pair of aligned words. The aligned words themselves are either *correct* (identical to original) or *substitutions* (different from the original).

We define word error rate as $\mathrm{WER} = (\mathrm{S} + \mathrm{I} + \mathrm{D})/(\mathrm{S} + \mathrm{I} + \mathrm{C})$ (I=insertion, D=deletion, S=substitution, C=correct). Note that this definition slightly deviates from the more commonly used $\mathrm{WER} = (\mathrm{S} + \mathrm{I} + \mathrm{D})/(\mathrm{S} + \mathrm{D} + \mathrm{C})$, which refers to the *reference* (in the denominator). Since we choose to refer to the hypothesis (the summary), we have to avoid a division by zero which could occur in cases where there are *only* insertions. (The two formulas yield very similar results in practice, and yield identical results if the number of insertions and deletions are balanced,

```
TURN 1:
rel:      0.5  0.5 0.5     0.5                0.75  0.75   ***
REF:      this is  to      illustrate         the   idea   ***
HYP:      this is  to      ILLUMINATE         ***   idea   AND
err:      C    C   C       S                  D     C      I
con:      1    1   1       0.9                -     0.8    0.8

TURN 2:
rel:      0   1    1  1       1       1          1
REF:      and here we have    very    relevant   information
HYP:      and HE   ** BEHAVES ****    IRREVERENT FORMATION
err:      C   S    D  S       D       S          S
con:      0.8 0.7  -  0.8     -       0.8        0.9
```

Figure 4.9
Simplified example of two turns (for score computation)

which is a goal sought for by speech recognition engines). The summary accuracy scores sa are defined as described in section 3.6.5. Relevance scores for insertions and substitutions are always 0.0.

To better illustrate how these metrics work, we demonstrate them on a simplified example of only two speaker turns (Figure 4.9).

The first line represents the relevance score r for each word (the number this word was within a relevant phrase divided by the number of annotators for that text). In turn 1, “this is to illustrate” was only marked relevant by two annotators, whereas “the idea” by 3 out of 4 annotators. The second line provides the reference transcript, the third line the ASR transcript. Line 4 gives the type of word error, and line 5 the confidence score of the speech recognizer (between 0.0 and 1.0, 1.0 meaning maximal confidence).

Now let us assume that turn 2 shows up in the summary. The scores are computed as follows:

- When summarizing the *reference*: Here, the word error rate is trivially 0.0; the summary accuracy sa is the sum of all relevance scores (=6.0) divided by the *maximal achievable score* with the same number of words ($n = 7$). Turn 2 has

6 words which were marked relevant by all coders ($r = 1.0$), turn 1's highest score is $r = 0.75$. Therefore: $sa_2 = 6.0/(6.0 + 0.75) = 0.89$. This is higher than the summary accuracy for turn 1: $sa_1 = 3.5/6.0 = 0.58(n = 6)$.

- When summarizing the *ASR transcript* (hypothesis): Selecting turn 2 will give $sa_2 = 0.0/2.25 = 0.0$ ($n = 5$). For turn 1, $sa_1 = 2.25/(0.75+0.5+0.5+0.5+0.0+0.0) = 1.0$ ($n = 6$; the sum in the denominator can only use relevance scores based on the *aligned* words w_a which were *correctly* recognized, therefore the 1.0-scores in turn 2 cannot be used). Turn 2 has WER=6/5=1.2, turn 1 has WER=3/6=0.5.

Obviously, when summarizing the ASR output, we would rather have turn 1 showing up in the summary than turn 2, because turn 2 is completely off from the truth and turn 1 only partially. The fact that turn 2 was considered to be more relevant by human coders cannot, in our opinion, be used to favor its inclusion in the summary. An exception would be a situation where the user has immediate access to the audio as well and is able to listen to selected passages from the summary. In our case, where we focus on text-only summaries to be used stand-alone, we have to minimize their word error rate. Given that, turn 1 has to be favored over turn 2, both because of its lower WER and because of its higher accuracy with respect to the relevance annotations.

4.3.3 Data characteristics

Tables 4.8 and 4.9 describe the main features of the corpus we used for our experiments: we selected seven dialogues from the `devtest`-set: 1 NewsHour, 2 CrossFire, and 4 CallHome dialogues. While the CallHome dialogues are narrow band (8kHz, channel=telephone), the TV shows were sampled at 16kHz (broadband). Both were automatically transcribed using a gender independent, vocal tract length normalized, large vocabulary speech recognizer which was trained on about 80 hours of Broadcast News data (Yu, Finke, and Waibel, 1999; Waibel et al., 2001) for the TV sub-corpus and on SwitchBoard/CallHome data, for the CallHome corpus. The word error rates for the 7 dialogues range from 30% to 60%.

	19CENT	BUCHANAN	GRAY
TV show	NewsHour	Crossfire	Crossfire
words in transcript	1310	3364	2307
words in ASR output	1373	3544	2034
topical segments	4	4	3
word error rate (in %)	32.6	32.6	58.5

Table 4.8
Characteristics of the broad-band sub-corpus (TV shows).

	EN_4335	EN_4371	EN_5573	EN_6161
corpus	CallFriend	CallFriend	CallHome	CallHome
words in transcript	2574	2570	2101	2232
words in ASR output	2476	2429	2082	2257
topical segments	5	5	5	6
word error rate (in %)	52.1	53.9	47.3	45.7

Table 4.9
Characteristics of the narrow-band sub-corpus (CallHome/CallFriend).

	19CENT	BUCHANAN	GRAY	EN_4335	EN_4371	EN_5573	EN_6161
(i) sum	-0.36	-0.26	-0.06	-0.41	-0.32	-0.38	-0.37
(ii) average	-0.44	-0.40	-0.07	-0.43	-0.36	-0.40	-0.36
(iii) norm. scores $> \theta$	-0.49	-0.46	-0.09	-0.54	-0.45	-0.43	-0.45
(iv) geometric mean	-0.40	-0.38	-0.06	-0.41	-0.31	-0.39	-0.32

Table 4.10
Pearson r correlation between WER and confidence scores

4.3.4 Word error rate reduction

In order to increase the likelihood that turns with lower WER are selected over turns with higher WER, we make use of the speech recognizer's confidence scores which are attached to every word hypothesis and can be viewed as probabilities: they are in [0.0,1.0], high values reflecting a high confidence in the correctness of the respective word.[4] Following (Valenza et al., 1999) we conjecture that we can use these confidence scores to increase the probability of passages with lower WER to show up in the summary. To test how far this assumption is justified, we correlated the WER with various metrics of confidence scores, computed in buckets of 20 words each: (i) sum of scores, (ii) average of scores, (iii) normalized number of scores above a threshold, and (iv) the geometric mean of scores. Table 4.10 shows the correlation coefficients (Pearson r) for the 7 ASR transcripts we used in our experiments.

Since we achieve the highest correlation coefficient (absolute value) for method (iii = avgth) (average number of words whose confidence score is greater than a threshold of 0.95), we apply this metric to the computation of turn-query similarities. We use the two following formulae to adjust the similarity-scores. (We shall call these adjustments MULT and EXP in the following.)

$$[mult] \quad sim_1' = sim_1(1 + \alpha \mathrm{avgth}) \tag{4.1}$$

$$[exp] \quad sim_1'' = sim_1 \mathrm{avgth}^{\alpha} \tag{4.2}$$

For both equations it holds that if $\alpha = 0.0$, the scores don't change, whereas if $\alpha > 0.0$, we enhance the weights of turns with many high confidence scores (*boost-*

[4] The speech recognizer computes these scores based on the acoustic stability of words during lattice rescoring.

	WER			summ.accuracy		
	full dialogue	$\alpha = 0$	$\alpha = 2$	human transcr.	$\alpha = 0$	$\alpha = 2$
en_4335	0.521	0.607	0.458	0.526	0.321	0.353
en_4371	0.539	0.454	0.377	0.449	0.313	0.322
en_5573	0.473	0.364	0.389	0.542	0.321	0.265
en_6161	0.457	0.392	0.264	0.529	0.386	0.471
19cent	0.326	0.265	0.265	0.577	0.474	0.474
buchanan	0.326	0.205	0.187	0.632	0.530	0.526
gray	0.585	0.405	0.448	0.546	0.307	0.323
average	0.461	0.385	0.341	0.543	0.379	0.391

Table 4.11
Effect of α on MMR baseline summary accuracy and WER (EXP method).

ing) and hence increase their likelihood of showing up earlier in the summary.[5]

Even though our evaluation method looks like it would guarantee an increase in summary accuracy when the word error rate is reduced, this is *not* necessarily the case. For example, it could turn out that while we can reduce WER by boosting passages with higher confidence scores, those passages might have (much) fewer words marked relevant than those being present in the summary without boosting. This way, it would be conceivable to create low word error summaries that contain also very few *relevant* pieces of information. As we will see later, while the latter is true for *some* dialogues, on average, WER reduction goes hand in hand with an increase of summary accuracy.

4.3.5 Experiments on automatically generated transcripts

We created summaries from ASR transcripts, using the optimized parameters from the general global evaluation of our system, depending on which sub-corpus a dialogue belongs to. Additionally to the summary accuracy, we evaluate now also the WER for each evaluation point. We varied α from 0.0 to 10.0 to see how much of an effect we would get from the boosting of turns with many high confidence scores (see equations 4.1 and 4.2). Both WER and summary accuracy scores are averaged over topical segments and the usual five summary sizes (5-25% sum-

[5]For EXP, we define $0^0 = 0$.

	WER			summ.accuracy		
	full dialogue	$\alpha = 0$	$\alpha = 2$	human transcr.	$\alpha = 0$	$\alpha = 2$
en_4335	0.521	0.590	0.406	0.618	0.349	0.424
en_4371	0.539	0.488	0.451	0.583	0.370	0.366
en_5573	0.473	0.394	0.371	0.579	0.393	0.362
en_6161	0.457	0.393	0.233	0.597	0.420	0.498
19cent	0.326	0.297	0.216	0.496	0.390	0.549
buchanan	0.326	0.297	0.238	0.604	0.480	0.452
gray	0.585	0.273	0.245	0.456	0.322	0.377
average	0.461	0.390	0.309	0.562	0.389	0.433

Table 4.12

Effect of α on DiaSumm summary accuracy and WER (EXP method).

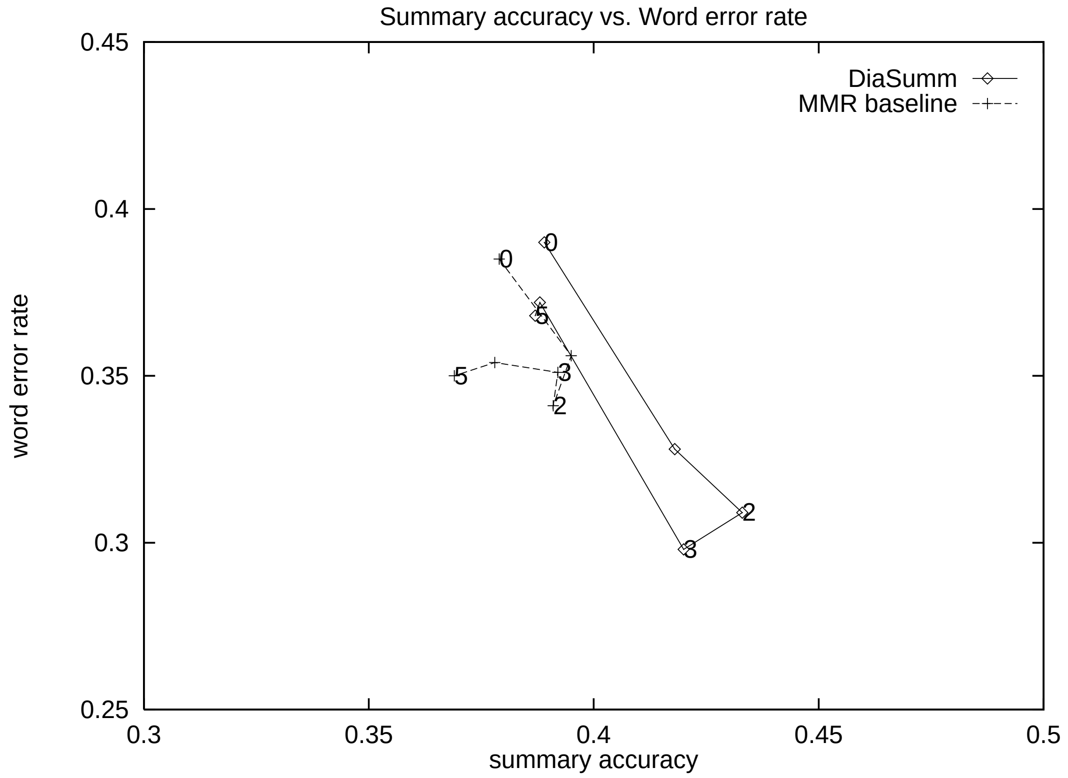

Figure 4.10

Summary accuracy vs. word error rates with EXP boosting ($0 \le \alpha \le 5$)

mary length in words).

Since the EXP formula yielded better results than MULT, we will only report on EXP-results. To compare the results of the two different methods at the different values of α, we compute the average difference between the baseline case ($\alpha = 0$, i.e., no confidence boosting) and the respective parameter settings, both for the summary accuracy as well as for WER. We then sum these two differences, taking the *negative* value of the WER-difference, since we are interested in WER reduction (but summary accuracy gain). We call this combination the *combined boosting benefit* *cbb*: $cbb = (sa_\alpha - sa_0) - (wer_\alpha - wer_0)$. For both MMR baseline and the full DiaSumm system, we find a maximum of *cbb* for $\alpha = 2.0$ (method EXP). For the full DiaSumm system, $cbb = 0.125$, for the MMR baseline the effect size is less than half: $cbb = 0.055$. The larger part of the effect is due to a WER reduction. Figure 4.10 shows the trade-off between WER and summary accuracy for the DiaSumm and MMR summaries, when α varies from 0 to 5.

In general, we see from Tables 4.11 and 4.12 that except for one dialogue, the typical summary already has a lower WER than the whole dialogue (over 15% relative reduction from .46 to .39), even without confidence boosting. This seems to indicate that in general, more relevant passages are also more likely to have fewer errors than less relevant passages. When we activate confidence boosting (with $\alpha = 2$), the average WER for DiaSumm dialogues is reduced by over 20% relative (from .390 to .309), and for the MMR baseline by over 11% (from .385 to .341).

Looking now at summary accuracy, we observe an average increase of over 11% relative from 0.389 to 0.433 for the DiaSumm summaries, and of over 3%, from 0.379 to 0.391, for the MMR baseline.

Table 4.13 shows the WER and sa differences for the 7 individual dialogues (all between $\alpha = 2$ and $\alpha = 0$) for the DiaSumm summaries. We see that the absolute WER reduction varies between 2 and 19%, and the absolute change in summary accuracy between -2 and +16%. While summary accuracy is reduced in 3 dialogues, only in one of these does this reduction outweigh the reduction in WER (en_5573, $cbb < 0$).

	summ.accuracy difference	WER difference
en_4335	0.075	-0.184
en_4371	-0.004	-0.037
en_5573	-0.031	-0.023
en_6161	0.078	-0.160
19cent	0.159	-0.081
buchanan	-0.028	-0.059
gray	0.055	-0.028

Table 4.13
Absolute differences in summary accuracy and WER between confidence boosting and baseline, for DiaSumm summaries.

4.3.6 Conclusion

The most significant result of these experiments is, in our opinion, the fact that the trade-off between *word error rate* and *summary accuracy* indeed leads to an optimal parameter setting for the creation of textual summaries for spoken language. Using a formula which emphasizes sentences containing many high confidence scores leads to an average WER reduction of over 20% for DiaSumm summaries and to an average improvement in summary accuracy of over 10%, compared to the baseline of a summary without using confidence boosting.

Comparing our results to those reported in (Valenza et al., 1999), we find that their relative WER reduction for summaries over full texts was considerably larger than ours (60% vs. 33%). We conjecture that reasons for this may be due to the different nature and quality of the confidence scores, and (not unrelated), to the different absolute WER of the two corpora (25% vs. 46%): in transcripts with higher WER, the confidence scores are usually less reliable.

4.4 Increasing the Local Summary Coherence by Cross-Speaker Information Linking

This section investigates the issue, how local coherence and informativeness of summaries change when we use our methods for automatic Q-A linking, described in section 3.4.

4.4.1 Introduction

Summary coherence is one of the major challenges for extract based summarization methods, where summaries are not generated from some abstract semantic representation, resulting from a deeper analysis of the original text, but rather composed of passages from the original text. When automatically summarizing well structured written texts, such as newswire data or scientific papers, the following strategies have been used to increase local coherence of such summaries, sometimes used in combination with each other: (a) lead based summaries (extracting the contiguous header of the text) (Brandow, Mitze, and Rau, 1995; Wasson, 1998); (b) paragraph based summaries (using paragraphs as minimal extraction units) (Mitra, Singhal, and Buckley, 1997; Salton et al., 1997); (c) inclusion of sentences which likely contain antecedents to anaphora in the current summary sentences (Johnson et al., 1993); (d) replacement of pronouns and definite descriptions by their antecedents (by means of automatic anaphora resolution) (Boguraev and Kennedy, 1997). Other methods, such as automatic analysis of discourse structure (Marcu, 2000), are more aimed at increasing the global coherence of a summary.

Spoken dialogue summarization introduces at least one additional dimension of coherence which is absent from written text generated by a single author: local cross-speaker coherence. Speakers accept or deny requests from each other, pose and answer questions, or acknowledge or comment on what was said by another dialogue participant. This section addresses the issue of the extent that cross-speaker information linking helps to increase the local coherence of spoken dialogue summaries and its effect on summary informativeness.

In this section, we use the unbalanced decision tree as question detection component, which yields the best compromise in overall performance (pr_{avg}) and runtime. Further, we do make use of the disfluency detection components, but use optimal sentence boundaries (from the human annotations).

	Q-A pairs	no Q-A det.	automatic	oracle
8E-CH	68	0.569 (0.170)	0.568 (0.169)	0.559 (0.170)
4E-CH	69	0.605 (0.128)	0.608 (0.123)	0.599 (0.139)
NHOUR	18	0.457 (0.232)	0.476 (0.248)	0.453 (0.230)
XFIRE	79	0.603 (0.129)	0.621 (0.151)	0.598 (0.118)
G-MTG	32	0.572 (0.194)	0.595 (0.155)	0.572 (0.194)
total	266	0.574 (0.163)	0.582 (0.163)	0.568 (0.164)

Table 4.14
Average summary accuracy (with standard deviations in brackets) for 15% summaries, using three different Q-A-detection methods.

4.4.2 Influence on summary accuracy

Summary generation using detected Q-A-regions

When we use the Q-A-detection component to aid summarization, the basic MMR algorithm stays the same. However, whenever a sentence which is part of a Q-A-region is put into the ranked list, the whole region is now added to the summary. This amounts to taking the maximum MMR score of the sentences within a Q-A-region to be its representative. Q-A regions are always described with the triple of sentence-IDs defined above: $< Q, A_{start}, A_{end} >$.

This subsection uses a numeric score, *summary accuracy*, to represent the quality of a summary. It is based on human relevance annotations of the dialogues and reflects how close the summary represents the opinion of the majority of the annotators (see section 3.6.5).

Experiments

To get an idea about how the summary accuracy changes using Q-A-pair detection and linking, we first tuned the parameters of the MMR summarization system, using the 8E-CH sub-corpus only. For the Q-A-detection component, we use three different options: (1) no Q-A-detection (this is the baseline system for this experiment), (2) automatic Q-A-pair detection with the unbalanced Q-detection decision tree and the A-detection script, and (3) optimal Q-A-pair detection, using an oracle

evaluation dimension	Informativeness			Fluency		
Q-A detection method	no	auto	oracle	no	auto	oracle
average score	3.18	3.18	3.24	2.82	3.12	3.50
average rank score	2.01	2.00	1.99	1.68	2.02	2.30

Table 4.15
Results of the user study comparing three different versions of summaries (average across all subjects and texts; $n = 66$).

informed by the human annotators' mark-ups.

Table 4.14 shows the results of these experiments. While we note that in most cases, the differences are rather small (t-test: $t < .6$, no significant differences overall), we have to take into account the low number of Q-A-pairs in most of the dialogues. In dialogues with a larger number of Q-A-pairs, there is sometimes a noticeable improvement in summary accuracy, particularly for the automatic Q-A detection method. On average, the accuracy scores for the oracle summaries are slightly below the baseline, while the summaries using our automatic detection module are slightly above the baseline. In short, this experiment shows that using Q-A-detection for summary generation does not significantly affect summary relevance.

4.4.3 User study

For the purpose of testing whether Q-A-detection can increase the local coherence of summaries, we performed a user study. We picked the 15 dialogue segments with the highest number of questions, since we wanted to quantify the effect of Q-A detection on texts which are particularly rich in Q-A-regions. For each of these dialogue segments, we took the same three versions of summaries described in the preceding subsection, each of them again at 15% length of the original (by word count). We had to exclude four segments which did not change when using Q-A-detection, due to the fact that the top-ranked sentences did not belong to any Q-A regions.

We then asked 6 subjects to rank the three different versions of summaries of the remaining 11 texts for (a) informativeness and (b) fluency (the latter should reflect local coherence). To aid the ranking process, the subjects had to score the summaries first using a discrete scale from 1 to 5 (for both dimensions). Informativeness should measure how much information the summary contains ("dense" vs. "sparse" text); the criterion for fluency should be how easy it is to read the summary and how coherent it is.

The order of the texts, as well as the summary versions within each text, were randomized. The average summary length was 142 words, thus each subject had to read and evaluate a text corpus of about 4700 words, which took, on average, about 31 minutes to complete. We provide the instructions for this user study, along with an example of a set of summaries, in Appendix D.1.

Table 4.15 presents the results of this study. Each number in the table is the average of 66 scores (11 texts times 6 subjects). For the rank scores, we gave 3 points to the first rank, 2 to the second, and 1 point to the last ranked summary version. In case of rank ties, we assigned 2.5 points (for rank 1=2) or 1.5 points (for rank 2=3), respectively. We observe that while the informativeness of the different summary versions does not change on average (no statistical difference), there is a significant improvement in fluency over the baseline for both summaries using automatic Q-A detection and oracle Q-A detection (significant at $\alpha < 0.05$, using the t-test). Individual subjects' scores did not differ much in these overall trends.

4.4.4 Discussion

Both the results from the automatic summary accuracy evaluation, as well as the results from the user study show that using Q-A detection does not significantly decrease the informativeness of the resulting summaries: neither evaluation showed a significant difference in information content or relevance for the three different versions of summaries. At the same time, as the user study clearly indicates, there is a significant benefit to be gained from including Q-A-regions in the summary in terms of summary fluency or local coherence.

We looked into the question why summary accuracy (on average) seems to be improving slightly (though not significantly) when we use our automatic Q-A detection module, while it stays at about the level of the baseline when using the oracle Q-A detection. When inspecting the summaries where this effect is most pronounced, we find that the main reason lies in the difference in Q-A region size between the automatic method and the oracle: While the oracle, derived from the human Q-A annotations, typically generates short answers ("core answers"), the automatic method tends to produce somewhat longer answers, consisting of multiple sentences. Particularly in cases where the core answer consists of only one or very few words (e.g., "yes"), the gain for summary accuracy is negligible. To avoid this effect, the oracle answer regions would probably have to be designed longer than they currently are, but this might have an adverse effect on Q-A detection training and testing accuracy. Another side effect of the shorter Q-A regions of the oracle method is that there are some (albeit few) cases where the MMR ranking module misses a Q-A region because the oracle Q-A region does not include the current MMR-selected sentence in its answer-part; this sentence is part of an extended answer region, which is in fact detected by the automatic answer detection module.

We also want to note some mostly genre-specific phenomena, which pose problems for the Q-A detection component:

- In CALLHOME, we sometimes encounter quoted questions, as in "A: he said: do you like it?" — "A: and i said: yes". The answer detection module fails here since the answer is provided by the speaker posing the (quoted) question.
- A similar case, also mostly in CALLHOME, is self-answered questions, such as "A: what is my plan?" — "A: to graduate next spring".
- In CROSSFIRE, we sometimes encounter questions with anaphoric reference to larger parts of the discourse, where the linking to their answers helps little for the local summary coherence (e.g., "would you accept that?", "do you agree to this?").

4.5 User Study

The purpose of the user study is to complement the findings of the automatic evaluation of the summarizer which uses the concept of summary accuracy. The focus is an intrinsic evaluation of summary informativeness, where users have to answer pre-determined multiple choice questions on different versions of summaries.

4.5.1 Data preparation

In the annotation phase, human coders had to select 3 questions for each topical segment which should reflect on core information present in their summary. To each question, the optimal answer was added, together with three multiple choice distractors. In the gold standard phase, the two involved annotators proceeded in the same way, matching three questions with their answers and multiple choice distractors for each topical segment. The target length for these gold standard summaries was 15% of the original text length; in practice, the lengths varied from about 7 to 22% with an average length of about 13% (nucleus-IUs only; there were very few satellites annotated for the gold standard).

For the user study, we picked all 34 topical segments from the 9 dialogues of the `eval`-set, yielding 102 questions to be answered. The sequence of topical segments was randomized, but every subject had the same sequence of topics in their experiment script. We divided the subjects into five groups since we want to compare the following five types of summaries, each with exactly matching length in words with the gold standard summaries, but to avoid to present one topic to a subject in more than one summary type:

1. LEAD: the first N words from the beginning of the topic
2. MMR: the baseline-MMR system, i.e., using the information condensation component only
3. DIASUMM: the full DIASUMM system with all its dialogue specific components
4. NPTELE: derived from DIASUMM, a telegraphic summary version, based mostly on noun phrases
5. gold: the human gold standard summary

Within every group, the sequence of summary types was also randomized, but of course we made sure that every topical segment was presented with every summary type once across the five groups.

Further we had to make sure that the formatting of the summaries was uniform so that the subjects should have minimal cues as to which summary was automatically vs. manually created.

Appendix D.2 provides the instructions for this user study along with an example of a set of 5 different summary types and the corresponding 3 multiple choice questions.

The third randomization concerns the order of the answers and the position of the correct answer in the list of four multiple choice options.

We note up front that our system was intentionally not trained to be able to cover the correct answers of multiple choice questions in either `devtest` or `eval` sub-corpora; all the training was focusing on coming close to the "majority opinion passages", which do not necessarily correspond exactly to the gold standard summaries, as can be seen by their typical scores in the .7-.8-range (Table 4.1 in section 4.1).

4.5.2 Experiment

We tested 25 subjects, 5 subjects per group. The average duration of the experiment was about 34.5 minutes per subject and 19.7 seconds per answer (median=20.8 seconds per answer).[6] Since the first answer includes the reading of the text, it took significantly longer than the second or third answers, where potentially only portions of the texts had to be re-read or checked (30.7/14.5/13.8 seconds on average).

As for correct answers, the average score (percent of correct answers) was 53.8 (median=54.9), compared to the random guessing baseline of 25.0. Figure 4.11 plots each subject's average answer accuracy vs. his/her average answer time. The correlation between answer time and score is only $r = 0.34$ which is barely significant at $p < 0.05$. However, excluding the subject with the lowest answer time and lowest score (ID=5-2), reduces the correlation coefficient for the remaining 24 subjects to only $r = 0.19$, which is not significant any more. We conclude therefore, that performance on the user study (in terms of answer accuracy) is largely independent of answer time.

When correlating all subjects' score vectors against each other, we find that the within-group correlation is far stronger than the correlation between different groups. Only 1 of the 50 within-group correlation coefficients is not significant

[6]The difference between 102*19.7 and 34.5*60 is explainable by the idle time between topical segments, where subjects had to click a button to be presented with the next text; these 33 empty intervals had an average duration of about 2 seconds each.

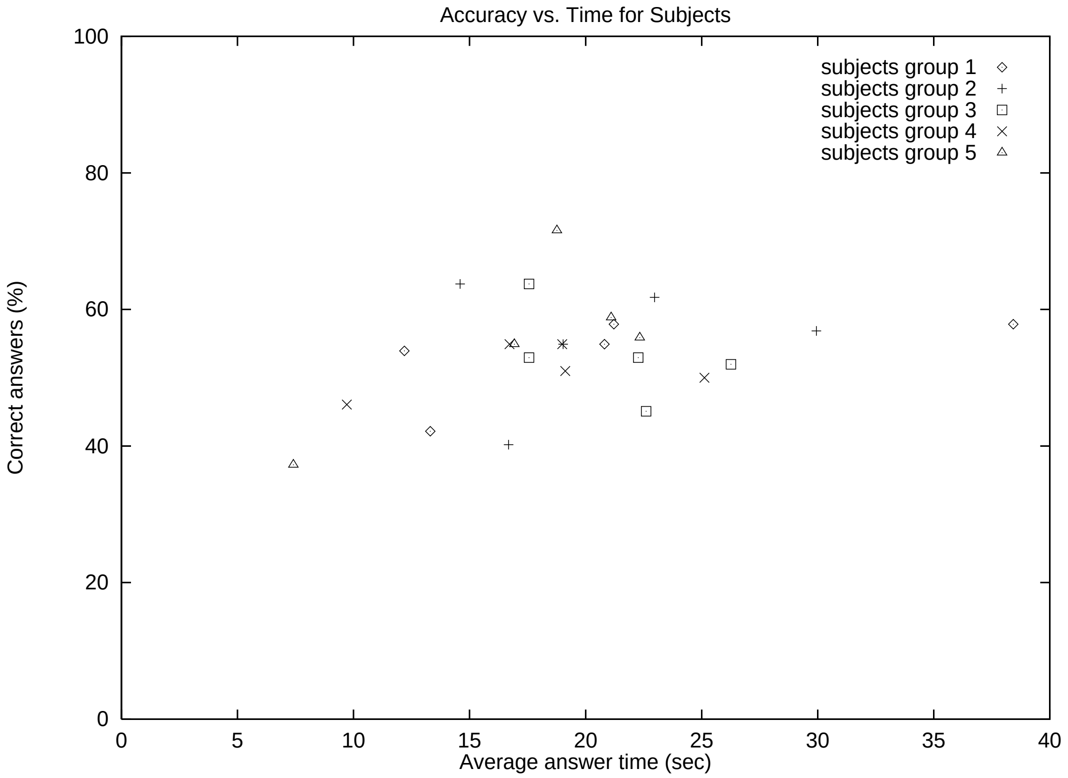

Figure 4.11
Answer accuracy vs. answer time for all 25 subjects of the user study.

	LEAD	MMR	DIASUMM	NPTELE	gold
4CH-EVAL	0.458	0.548	0.484	0.490	0.826
evalNH	0.500	0.400	0.233	0.533	0.867
evalXF	0.400	0.438	0.400	0.495	0.714
evalMtg	0.300	0.467	0.633	0.400	0.600

Table 4.16
Average answer scores for five different summary types over four different sub-corpora.

	LEAD	MMR	DIASUMM	NPTELE	gold
4CH-EVAL	18.179	17.643	17.645	19.315	14.058
evalNH	27.336	16.519	26.835	23.413	22.291
evalXF	27.985	26.386	24.305	27.100	22.732
evalMtg	25.816	20.777	20.609	20.083	19.468

Table 4.17
Average answer times (in seconds) for five different summary types over four different sub-corpora.

at the 0.05-level, and 47 of 50 coefficients are significant at the 0.01-level. This indicates that within a group (meaning: subjects with identical experiment set-ups), the outcomes were fairly consistent.

Table 4.16 and 4.17 show the results (relative number of correctly answered questions and seconds to answer the questions), broken down by sub-corpus and summary type.

The scores of the automatic summary accuracy evaluation for the same topical segments (all the same length, matching the gold standard summaries with nucleus-IUs only) are shown in Table 4.18.

Figures 4.12 and 4.13 show the comparison of automatic scores and results from the user study for the CALLHOME and the CROSSFIRE sub-corpora, respectively.[7] We added two more scores here derived from an optimal answer key for every question for every type of summary text in the user study (3*5*34=510 questions total). One score is generated on the basis of both "sure answers" (answer is explicitly contained in the summary) as well as "likely answers" (where the correct

[7]The other two corpora have only 2 topical segments each.

	LEAD	MMR	DIASUMM	NPTELE	gold
4CH-EVAL	0.424	0.503	0.590	0.567	0.793
evalNH	0.682	0.552	0.461	0.547	0.850
evalXF	0.403	0.538	0.580	0.533	0.790
evalMtg	0.332	0.465	0.606	0.631	0.704

Table 4.18
Summary accuracy scores of five different summary types across four sub-corpora (same length in words).

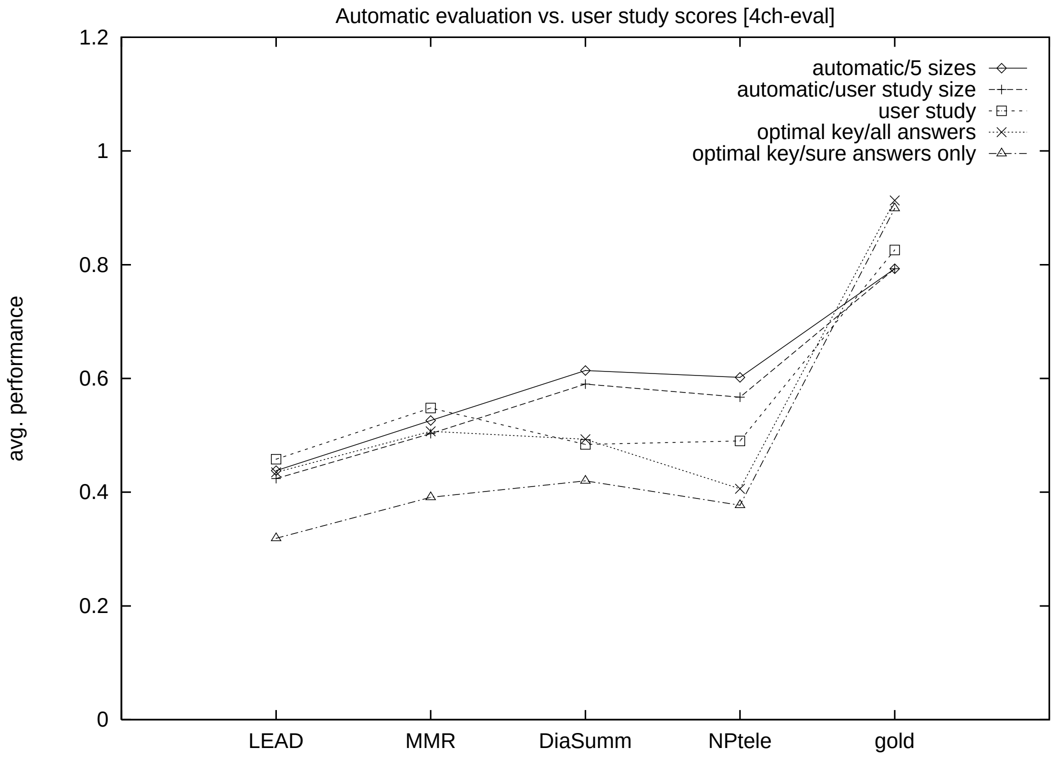

Figure 4.12
Comparison of automatic evaluation results, user study results, and optimal answer keys (2 variants) for the 4CH-EVAL sub-corpus.

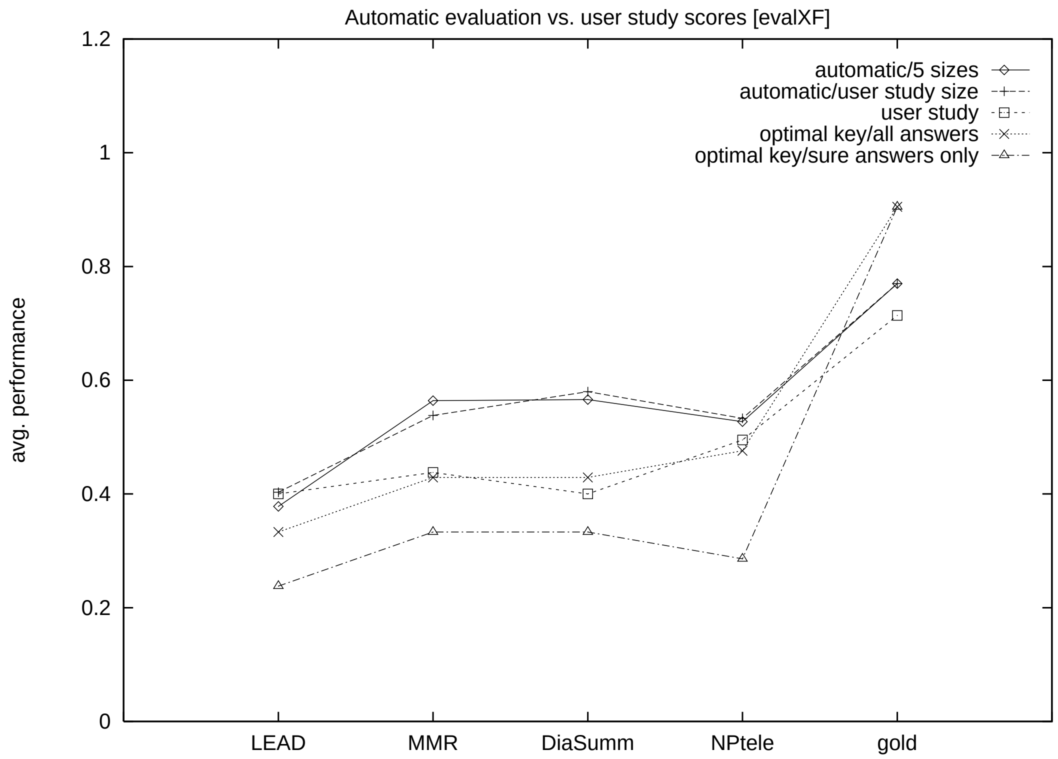

Figure 4.13
Comparison of automatic evaluation results, user study results, and optimal answer keys (2 variants) for the XFire sub-corpus.

answer can be guessed from context or general knowledge) (`optkey-all`). The other score, `optkey-strict`, only considers the first type of answers: "sure answers". (For answers not to be found or to be guessed, a dummy answer is assumed corresponding to a random guess.) Note that the y-axis of these two figures folds two different scores into one scale: summary accuracy for automatic evaluations and fraction of correctly answered questions for evaluations pertaining to the user study.

4.5.3 Discussion

As for average answer time, the trends are mixed,[8] but we observe that usually, LEAD and NPTELE summaries take a bit longer to answer than both the MMR baseline and the DIASUMM summaries. Further, the gold summaries usually have the shortest answer times. As for LEAD and gold summaries, we think that their respective speed difference is mostly due to the fact that one scores worst on average and the other best, and so users might take a longer time to look for an answer which is not to be found in LEAD summaries, whereas in gold summaries, the answers are readily available. The case for NPTELE maybe slightly different, in that it deviates strongest from the usual sequential text format and only presents scattered pieces of information which may well take longer to read and to process than normal text.

As for answer scores, the overall trends for the automatic evaluation look very similar to the `optkey-strict` case, reflecting the answers actually contained in the summaries. Since the subjects were able (and encouraged) to guess their answers in cases where it was not to be found in the summary text, these basic results get slightly distorted, reflected in the similarity of the actual users' scores and the `optkey-all` case.

When discussing the answer scores in the following, we have to keep in mind that most of the differences are not significant, mostly due to the small number of topical segments (except for the 4CH-EVAL sub-corpus). Exceptions are (a) the differences between the gold standard summaries and the four types of automatically generated summaries (4CH-EVAL: both automatic evaluation and user

[8] In fact, the only significant difference in this table is that the answer times for 4CH-EVAL-gold-summaries are significantly lower than for the four other summary types of that sub-corpus.

study scores; CROSSFIRE: only automatic evaluation scores), (b) the disadvantage of LEAD compared to the four other summary types (4CH-EVAL: automatic evaluation scores), and (c) the advantage of DIASUMM over MMR summaries in 4CH-EVAL (automatic evaluation). (All significance tests were performed with the Wilcoxon signed rank sum test.)

While for the EVALMTG-corpus, DIASUMM is better than the MMR-baseline, for 4CH-EVAL, the trend is reversed (but not statistically significant). Further, for the two more formal corpora, NEWSHOUR and CROSSFIRE, the NPTELE summaries perform best (unlike for the automatic evaluation, where LEAD and DIASUMM performed best, respectively). This result suggests that the orthogonal text reduction (within-sentence) is most helpful when summarizing texts with longer and more complex sentences; while these summaries might not match our answer key (in the automatic evaluation) as well as the MMR or DIASUMM summaries, they enable a user to find the correct answer to key questions in the texts more often.

Surface structure is to deep structure
Observation is to speculation
As science is to mysticism.
Poster in a seminar room
of an academic linguistics department

Chapter 5

Conclusion

5.1 Discussion and Directions for Future Work

The problem of how to automatically generate readable and concise summaries for spoken dialogues in unrestricted domains has many challenges that need to be addressed. Some of the research issues are similar or identical to those faced when summarizing written texts (such as topic segmentation, determining the most salient/relevant information, anaphora resolution, summary evaluation), but other additional dimensions are added on top of this list, including speech disfluency detection, sentence boundary detection, cross-speaker information linking, and coping with imperfect speech recognition.

In this thesis, we decided to focus on the three problems of (i) speech disfluency detection, (ii) sentence boundary detection, and (iii) cross-speaker information linking, in addition to (iv) topic segmentation and (v) relevance ranking (MMR). We implemented the spoken dialogue system DIASUMM whose components address each of these issues and are trainable. Both the evaluations of the individual components of our spoken dialogue summarization system, as well as the global evaluations, have shown that we can successfully make use of the disfluency annotated SWITCHBOARD corpus (LDC, 1999b) to train the spoken language specific components of our system, in particular for genres of informal dialogues (CALLHOME and GROUP MEETINGS), where DIASUMM performs significantly better than a LEAD and a MMR baseline. We conjecture that the reasons why the DIASUMM system was not able to improve over the MMR baseline for the two other, more formal corpora, lies in their very nature of being of a quite different genre: the NEWSHOUR and CROSSFIRE corpora have longer turns and sentences, as well as fewer disfluencies, along with more complex sentence structures than typically found in the other corpora of more colloquial, spontaneous conversations. Future work will have to address the issue of whether the availability of training data for more formal dialogues (in size and annotation style comparable to the SWITCHBOARD corpus, though) could lead to an improvement in performance on those data sets, as well, or if even then a standard written text based summarizer would be hard to improve upon.

Given the complexity and size of our undertaking, we had to make a number of simplifying assumptions, most notably about the input data for our system: We use perfect transcriptions by humans instead of ASR transcripts (except for section 4.3), which, for these genres, typically show word error rates (WER) ranging from at least 10% to 50%. We have shown in this thesis, however, that the actual word error rates in summaries generated from ASR output are usually lower than the full transcript WER, and can further be significantly reduced by taking acoustically derived confidence scores into account.

Somewhat related to this is the fact that we did not look at the question whether and how much prosodic information such as stress or pitch contours could improve the individual components of our system, as well as its global performance. Past work in related fields (Stolcke et al., 1998; Stolcke et al., 2000) suggest that some improvements may indeed be achievable, but it is hard to predict the effect size for our particular task. We also conjecture that the role of prosodic information in automatic dialogue summarization will be a function of the correctness of the ASR output: for fully spontaneous, open domain, multi-party conversations in adverse recording environments (e.g., single table microphone, large and changing background noise, foreign accented speech, high cross-talk ratio), word error rates of speech recognizers will remain fairly high in the foreseeable future (currently, they are in the 40% WER range). For these and similarly challenging situations, the judicious use of prosodic information might help both for sub-tasks such as sentence boundary detection or topic detection, as well as for overall summary accuracy. Other sources of input to a summarization system, if available, might be of benefit of well, such as the output of focus tracking devices (face and gaze tracking) which might reflect the participants' attention to more or less important parts of the conversation.

Finally, we think that there is ample room for research in the area of automatically deriving discourse structures for spoken dialogues in unrestricted domains, even if the covered text spans might only be local (due to a lack of global discourse plans). We believe that a summarizer, in addition to knowing about the interactively constructed and coherent pieces of information (such as in question-answer pairs), could make good use of such structured information and be better guided in making its selections for summary generation. In addition, this discourse structure might facilitate modules which perform automatic anaphora detection and resolution.

5.2 Conclusions

This thesis represents a first attempt into investigating the challenge of automatically summarizing multi-party dialogues in diverse genres without any restriction on domain. We identify and address most of the major issues involved when dealing with spoken as opposed to written language, and with dialogues as opposed to monological texts.

We motivated, implemented, and evaluated an approach to automatically create extract summaries for open domain spoken dialogues in two informal and two formal genres of multi-party conversations. Our dialogue summarization system DIASUMM uses trainable components to detect and remove speech disfluencies (making the output more readable and less noisy), to determine sentence boundaries (creating suitable text spans for summary generation), to link cross-speaker information units (allowing for increased summary coherence), and to determine topically coherent regions in the dialogues.

We used a corpus of 23 dialogue excerpts from four different genres (80 topical segments, about 47000 words, about 4 hours of recorded speech) for system development and evaluation and the disfluency annotated SWITCHBOARD corpus (LDC, 1999b) for training of the three dialogue specific components. Our corpus was annotated by six human coders for topical boundaries and relevant text spans for summaries. Additionally, annotations were made for disfluencies and question speech acts and their answers. We devised a word-based evaluation metric, relative summary accuracy, which is meant to reflect how close the automatic summaries match relevant passages marked by human annotators.

In a global system evaluation we compared two baseline systems (LEAD and MMR) with the DIASUMM system using all of its dialogue specific components discussed in this thesis, using text segments with optimal topical boundaries as summarizer input. The results showed that (a) both the baseline MMR system as well as DIASUMM create better summaries than a LEAD baseline (except for the NEWSHOUR sub-corpus), and that (b) DIASUMM performs significantly better than the baseline MMR system for the informal dialogue corpora (CALLHOME and GROUP MEETINGS).

A user study was performed which tested the answer time and accuracy of a set of multiple choice questions for two different types of DIASUMM summaries (standard and NP-telegraphic), a human gold standard, and two baselines (MMR, LEAD). The study confirmed the trends of the automatic evaluation, but did not show a significant difference between the MMR baseline and the DIASUMM system. Further, the noun phrase summaries outperformed the standard DIASUMM summaries, as well as the two baselines, for the two formal sub-corpora (NEWSHOUR, CROSSFIRE).

With respect to automatic speech recognizer transcripts, we showed that we can make effective use of the speech recognizer's confidence scores to focus the summaries on passages with a higher likelihood of being correctly recognized. That way, on average, the word error rate decreased by more than 20% relative, while summary accuracy improved by over 11% relative.

Finally, the system was integrated in the *Meeting Browser*, a multi-media and multi-modal tool for recording, transcribing, archiving, searching, summarizing, and displaying of multi-party conversations. It allows for drill-down summarization, query-dependent summary generation, and lets the user choose from a set of different summary types. The typical run time of DIASUMM is about 1% of real time on a 900 MHz PentiumIII.

Bibliography

Alexandersson, Jan and Peter Poller. 1998. Towards multilingual protocol generation for spontaneous speech dialogues. In *Proceedings of the INLG-98, Niagara-on-the-lake, Canada, August*.

Altomari, Patrick J. and Patricia A. Currier. 1996. Focus of TIPSTER phases I and II. In *Proceedings of the TIPSTER text program phase II workshop*, pages 9–11.

Aone, Chinatsu, Mary Ellen Okurowski, and James Gorlinsky. 1997. Trainable, scalable summarization using robust NLP and machine learning. In *ACL/EACL-97 Workshop on Intelligent and Scalable Text Summarization, Madrid, Spain*.

Arons, Barry. 1994. Pitch-based emphasis detection for segmenting speech. In *Proceedings of the ICSLP-94*, pages 1931–1934.

Banko, Michele, Vibhu O. Mittal, and Michael J. Witbrock. 2000. Headline generation based on statistical translation. In *Proceedings of the 38th Conference of the Association for Computational Linguistics, Hongkong, China, October*, pages 318–325.

Barzilay, Regina and Michael Elhadad. 1997. Using lexical chains for text summarization. In *ACL/EACL-97 Workshop on Intelligent and Scalable Text Summarization, Madrid, Spain*.

Beeferman, Doug, Adam Berger, and John Lafferty. 1999. Statistical models for text segmentation. *Machine Learning*, 34(1-3):177–210.

Berger, Adam L. and Vibhu O. Mittal. 2000. OCELOT: a system for summarizing web pages. In *Proceedings of the 23rd ACM-SIGIR Conference*.

Bett, Michael, Ralph Gross, Hua Yu, Xiaojin Zhu, Yue Pan, Jie Yang, and Alex Waibel. 2000. Multimodal meeting tracker. In *Proceedings of the Conference on Content-Based Multimedia Information Access, RIAO-2000, Paris, France, April*.

Boguraev, Branimir and Christopher Kennedy. 1997. Salience-based characterisation of text documents. In *ACL/EACL-97 Workshop on Intelligent and Scalable Text Summarization, Madrid, Spain*.

Brandow, Ronald, Karl Mitze, and Lisa F. Rau. 1995. Automatic condensation of electronic publications by sentence selection. *Information Processing and Management*, 31(5):675–685.

Brill, Eric. 1994. Some advances in transformation-based part of speech tagging. In *Proceeedings of AAAI-94*.

Carbonell, Jaime, Yibing Geng, and Jade Goldstein. 1997. Automated query-relevant summarization and diversity-based reranking. In *Proceedings of the IJCAI-97 workshop on AI and digital libraries, Nagoya, Japan*.

Carbonell, Jaime and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st ACM-SIGIR International Conference on Research and Development in Information Retrieval, Melbourne, Australia*.

Carletta, Jean. 1996. Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics*, 22(2):249–254, June.

Chen, Francine R. and Margaret Withgott. 1992. The use of emphasis to automatically summarize a spoken discourse. In *Proceedings of the ICASSP-92*, pages 229–332.

Choi, Freddy Y. Y. 2000. Advances in domain independent linear text segmentation. In *Proceedings of ANLP-NAACL-2000, Seattle, WA, May*, pages 26–33.

Cohen, Jacob. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Davies, Mark and Joseph L. Fleiss. 1982. Measuring agreement for multinomial data. *Biometrics*, 38:1047–1051, December.

Edmundson, H. P. 1969. New methods in automatic extracting. *Journal of the ACM*, 16(2):264–285, April.

Garofolo, John S., Ellen M. Voorhees, Cedric G. P. Auzanne, and Vincent M. Stanford. 1999. Spoken document retrieval: 1998 evaluation and investigation of new metrics. In *Proceedings of the ESCA workshop: Accessing information in spoken audio*, pages 1–7. Cambridge, UK, April.

Garofolo, John S., Ellen M. Voorhees, Vincent M. Stanford, and Karen Sparck Jones. 1997. TREC-6 1997 spoken document retrieval track overview and results. In *Proceedings of the 1997 TREC-6 Conference, Gaithersburg, MD, November*, pages 83–91.

Gavaldà, Marsal. 2000. SOUP: A parser for real-world spontaneous speech. In *Proceedings of the 6th International Workshop on Parsing Technologies (IWPT-2000), Trento, Italy, February*.

Gavaldà, Marsal, Klaus Zechner, and Gregory Aist. 1997. High performance segmentation of spontaneous speech using part of speech and trigger word information. In *Proceedings of the 5th ANLP Conference, Washington DC*, pages 12–15.

Gee, F. Ruth. 1996. TIPSTER phase III goals. In *Proceedings of the TIPSTER text program phase II workshop*, pages 1–8.

Godfrey, J. J., E. C. Holliman, and J. McDaniel. 1992. SWITCHBOARD: telephone speech corpus for research and development. In *Proceedings of the ICASSP-92*, volume 1, pages 517–520.

Goldstein, Jade, Vibhu Mittal, Jaime Carbonell, and Mark Kantrowitz. 2000. Multi-document summarization by sentence extraction. In *Proceedings of the NAACL-ANLP-2000 Workshop on Automatic Summarization, Seattle, WA, April*.

Grosz, Barbara J. and Candace L. Sidner. 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Hearst, Marti A. 1997. TextTiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64, March.

Heeman, Peter A. and James F. Allen. 1999. Speech repairs, intonational phrases, and discourse markers: Modeling speakers' utterances in spoken dialogue. *Computational Linguistics*, 25(4):527–571, December.

Hirschberg, Julia and Christine Nakatani. 1998. Acoustic indicators of topic segmentation. In *Proceedings of the ICSLP-98, Sydney, Australia*.

Hirschberg, Julia, Steve Whittaker, Don Hindle, Fernando Pereira, and Amit Singhal. 1999. Finding information in audio: A new paradigm for audio browsing/retrieval. In *Proceedings of the ESCA workshop: Accessing information in spoken audio*, pages 117–122. Cambridge, UK, April.

Hori, Chiori and Sadaoki Furui. 2000. Automatic speech summarization based on word significance and linguistic likelihood. In *Proceedings of ICASSP-00, Instanbul, Turkey, June*, pages 1579–1582.

Hovy, Eduard and ChinYew Lin. 1997. Automated text summarization in SUMMARIST. In *ACL/EACL-97 Workshop on Intelligent and Scalable Text Summarization, Madrid, Spain*.

Hovy, Eduard and Daniel Marcu. 1998. Automated text summarization. Tutorial Notes, COLING-ACL 98, August 98.

Jing, Hongyan. 2000. Sentence reduction for automatic text summarization. In *Proceedings of ANLP-NAACL-2000, Seattle, WA, May*, pages 310–315.

Johnson, Frances C., Chris D. Paice, William J. Black, and A.P. Neal. 1993. The application of linguistic processing to automatic abstract generation. *Journal of Document and Text Management*, 1(3):215–241.

Kameyama, M. and I. Arima. 1994. Coping with aboutness complexity in information extraction from spoken dialogues. In *Proceedings of the ICSLP 94, Yokohama, Japan*, pages 87–90.

Kameyama, Megumi, Goh Kawai, and Isao Arima. 1996. A real-time system for summarizing human-human spontaneous spoken dialogues. In *Proceedings of the ICSLP-96*, pages 681–684.

Knight, Kevin and Daniel Marcu. 2000. Statistics-based summarization — step one: Sentence compression. In *Proceedings of the 17th National Conference of the AAAI*.

Koumpis, Konstantinos and Steve Renals. 2000. Transcription and summarization of voicemail speech. In *Proceedings of ICSLP-00, Beijing, China, October*, pages 688–91.

Kozima, Hideki. 1993. Text segmentation based on similarity between words. In *Proceedings of the 31st Annual Meeting of the ACL (student session)*, pages 286–288.

Kupiec, J., J. Pedersen, and F. Chen. 1995. A trainable document summarizer. In *Proceedings of the 18th ACM-SIGIR Conference*, pages 68–73.

Lavie, Alon, Alex Waibel, Lori Levin, Michael Finke, Donna Gates, Marsal Gavaldà, Torsten Zeppenfeld, and Puming Zhan. 1997. Janus III: Speech-to-speech translation in multiple languages. In *IEEE International Conference on Acoustics, Speech and Signal Processing, Munich, 1997*.

LDC, Linguistic Data Consortium. 1996. CallHome and CallFriend LVCSR databases.

LDC, Linguistic Data Consortium. 1999a. Addemdum to the Part-of-Speech Tagging Guidelines for the Penn Treebank Project (Modifications for thee SwitchBoard corpus). Linguistic Data Consortium (LDC) CD-ROM LDC99T42.

LDC, Linguistic Data Consortium. 1999b. Treebank-3: CD-ROM containing databases of disfluency annotated Switchboard transcripts (LDC99T42).

Levin, Lori, Klaus Ries, Ann Thymé-Gobbel, and Alon Lavie. 1999. Tagging of speech acts and dialogue games in spanish call home. In *Proceedings of the ACL-99 Workshop on Discourse Tagging, College Park, MD*.

Litman, Diane L. and Rebecca J. Passonneau. 1995. Combining multiple knowledge sources for discourse segmentation. In *Proceedings of the ACL-95*, pages 108–115.

Luhn, H. P. 1958. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165.

Mani, Inderjeet, David House, Gary Klein, Lynette Hirschman, Leo Obrst, Therese Firmin, Michael Chrzanowski, and Beth Sundheim. 1998. The TIPSTER SUMMAC text summarization evaluation. Mitre Technical Report MTR 98W0000138, October 1998.

Mani, Inderjeet and Mark T. Maybury, editors. 1999. *Advances in automatic text summarization*. MIT Press, Cambridge, MA.

Marcu, Daniel. 1997. From discourse structure to text summaries. In *ACL/EACL-97 Workshop on Intelligent and Scalable Text Summarization, Madrid, Spain*.

Marcu, Daniel. 1999. Discourse trees are good indicators of importance in text. In Mani and Maybury (Mani and Maybury, 1999), pages 123–136.

Marcu, Daniel. 2000. The rhetorical parsing of unrestricted texts: A surface-based approach. *Computational Linguistics*, 26(3):395–448.

Meteer, Marie, Ann Taylor, Robert MacIntyre, and Rukmini Iyer. 1995. Dysfluency annotation stylebook for the Switchboard corpus. Revised by Ann Taylor, June 1995, available on the LDC99T42 CD-ROM, published by LDC.

Miike, Seiji, Etuso Itoh, Kenji Onon, and Kazuo Sumita. 1994. A full-text retrieval system with a dynamic abstract generation function. In *Proceedings of the 17th ACM-SIGIR Conference*, pages 318–327.

Miller, George A., Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine Miller. 1993. Five papers on WordNet. Technical report, Princeton University, CSL, revised version, August.

Miller, George A., Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine Miller. 1990. Five papers on WordNet. Technical report, Princeton University, July.

Mitra, Mandar, Amit Singhal, and Christ Buckley. 1997. Automatic text summarization by paragraph extraction. In *ACL/EACL-97 Workshop on Intelligent and Scalable Text Summarization, Madrid, Spain*.

Morris, Jane and Graeme Hirst. 1991. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics*, 17(1):21–48.

Nakatani, Christine H. and Julia Hirschberg. 1994. A corpus-based study of repair cues in spontaneous speech. *Journal of the Acoustic Society of America*, 95(3):1603–1616, March.

NIST. 2001. Document Understanding Conference (DUC) 2001. http://www-nlpir.nist.gov/projects/duc/.

Quinlan, J. Ross. 1992. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA.

Radev, Dragomir R., Hongyan Jing, and Malgorzata Budzikowska. 2000. Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *Proceedings of the NAACL-ANLP-2000 Workshop on Automatic Summarization, Seattle, WA, April*, pages 21–30.

Radev, Dragomir R. and Kathleen R. McKeown. 1998. Generating natural language summaries from multiple on-line sources. *Computational Linguistics*, 24(3):469–500, September.

Rath, G. J., A. Resnick, and T. R. Savage. 1961. The formation of abstracts by the selection of sentences. *American Documentation*, 12(2):139–143.

Reimer, Ulrich and Udo Hahn. 1997. A formal model of text summarization based on condensation operators of a terminological logic. In *ACL/EACL-97 Workshop on Intelligent and Scalable Text Summarization, Madrid, Spain*.

Reithinger, Norbert, Michael Kipp, Ralf Engel, and Jan Alexandersson. 2000. Summarizing multilingual spoken negotiation dialogues. In *Proceedings of the 38th Conference of the Association for Computational Linguistics, Hongkong, China, October*, pages 310–317.

Ries, Klaus, Lori Levin, Liza Valle, Alon Lavie, and Alex Waibel. 2000. Shallow discourse genre annotation in CallHome Spanish. In *Proceedings of the Second Conference on Language Resources and Evaluation, LREC-2000, Athens, Greece, May/June*.

Rose, Ralph Leon. 1998. *The communicative value of filled pauses in spontaneous speech*. Ph.D. thesis, University of Birmingham, Birmingham, UK.

Salton, Gerard, editor. 1971. *The SMART Retrieval System — Experiments in Automatic Text Processing*. Prentice Hall, Englewood Cliffs, New Jersey.

Salton, Gerard, James Allan, Chris Buckley, and Amit Singhal. 1994. Automatic analysis, theme generation and summarization of machine-readable texts. *Science*, 264:1421–1426.

Salton, Gerard and Michael J. McGill. 1983. *Introduction to Modern Information Retrieval*. McGraw Hill, Tokyo etc.

Salton, Gerard, Amit Singhal, Mandar Mitra, and Chris Buckley. 1997. Automatic text structuring and summarization. *Information Processing and Management*, 33(2):193–207.

Santorini, Beatrice. 1990. Part-of-Speech Tagging Guidelines for the Penn Treebank Project. Linguistic Data Consortium (LDC) CD-ROM LDC99T42.

Shriberg, Elizabeth, Andreas Stolcke, Dilek Hakkani-Tür, and Gökhan Tür. 2000. Prosody-based automatic segmentation of speech into sentences and topics. *Speech Communication*, 32(1-2).

Shriberg, Elizabeth E. 1994. *Preliminaries to a Theory of Speech Disfluencies*. Ph.D. thesis, University of Berkeley, Berkeley, CA.

Sparck-Jones, Karen and Brigitte Endres-Niggemeyer. 1995. Automatic summarizing. *Information Processing and Management*, 31(5):625–630.

Stifelman, Lisa J. 1995. A discourse analysis approach to structured speech. In *AAAI-95 Spring Sympoisium on Empirical Methods in Discourse Interpretation and Generation, Stanford, CA, March*.

Stolcke, Andreas, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3):339–373, September.

Stolcke, Andreas and Elizabeth Shriberg. 1996. Automatic linguistic segmentation of conversational speech. In *Proceedings of the ICSLP-96*, pages 1005–1008.

Stolcke, Andreas, Elizabeth Shriberg, Rebecca Bates, Mari Ostendorf, Dilek Hakkani, Madeleine Plauche, Gökhan Tür, and Yu Lu. 1998. Automatic detection of sentence boundaries and disfluencies based on recognized words. In *Proceedings of the ICSLP-98, Sydney, Australia, December*, volume 5, pages 2247–2250.

Sundheim, Beth M. 1996. The message understanding conferences. In *Proceedings of the TIPSTER text program phase II workshop*, pages 1–8.

Teufel, Simone and Marc Moens. 1997. Sentence extraction as a classification task. In *ACL/EACL-97 Workshop on Intelligent and Scalable Text Summarization, Madrid, Spain*.

Valenza, Robin, Tony Robinson, Marianne Hickey, and Roger Tucker. 1999. Summarisation of spoken audio through information extraction. In *Proceedings of the ESCA workshop: Accessing information in spoken audio*, pages 111–116. Cambridge, UK, April.

van Mulbregt, P., I. Carp, L. Gillick, S. Lowe, and J. Yamron. 1998. Text segmentation and topic tracking on Broadcast News via a Hidden Markov Model approach. In *Proceedings of ICSLP-98, Sydney, Australia*.

Wahlster, Wolfgang. 1993. Verbmobil — translation of face-to-face dialogs. In *Proceedings of MT Summit IV, Kobe, Japan*.

Waibel, Alex, Michael Bett, and Michael Finke. 1998. Meeting browser: Tracking and summarizing meetings. In *Proceedings of the DARPA Broadcast News Workshop*.

Waibel, Alex, Michael Bett, Florian Metze, Klaus Ries, Thomas Schaaf, Tanja Schultz, Hagen Soltau, Hua Yu, and Klaus Zechner. 2001. Advances in automatic meeting record creation and access. In *Proceedings of ICASSP-2001, Salt Lake City, UT, May*.

Ward, Wayne. 1991. Understanding spontaneous speech: The PHOENIX system. In *Proceedings of ICASSP-91*, pages 365–367.

Wasson, Mark. 1998. Using leading text for news summaries: Evaluation results and implications for commercial summarization approaches. In *Proceedings of COLING/ACL-98, Montreal, Canada*, pages 1364–1368.

Whittaker, Steve, Julia Hirschberg, John Choi, Don Hindle, Fernando Pereira, and Amit Singhal. 1999. SCAN: Designing and evaluating user interfaces to support retrieval from speech archives. In *Proceedings of the 22nd ACM-SIGIR International Conference on Research and Development in Information Retrieval, Berkeley, CA, August*, pages 26–33.

Youmans, Gilbert. 1991. A new tool for discourse analysis: The vocabulary-management profile. *Language*, 67(4):763–789.

Yu, Hua, Michael Finke, and Alex Waibel. 1999. Progress in automatic meeting transcription. In *Proceedings of EUROSPEECH-99, Budapest, Hungary, September*.

Zechner, Klaus. 1997. Building chunk level representations for spontaneous speech in unrestricted domains: The CHUNKY system and its application to reranking N-best lists of a speech recognizer. Master's thesis (project report), CMU, available from: `http://www.cs.cmu.edu/~zechner/publications.html`.

Zechner, Klaus and Alon Lavie. 2001. Increasing the coherence of spoken dialogue summaries by cross-speaker information linking. In *Proceedings of the NAACL-01 Workshop on Automatic Summarization, Pittsburgh, PA, June*, pages 22–31.

Zechner, Klaus and Alex Waibel. 2000a. DIASUMM: Flexible summarization of spontaneous dialogues in unrestricted domains. In *Proceedings of COLING-2000, Saarbrücken, Germany, July/August*, pages 968–974.

Zechner, Klaus and Alex Waibel. 2000b. Minimizing word error rate in textual summaries of spoken language. In *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics, NAACL-2000, Seattle, WA, April/May*, pages 186–193.

Appendix A

List of POS Tags

```
TAG  Penn-Treebank-Number/POS tag description
---------------------------------------------
''   direct speech marker
BES  SWBD-special: BE-s ('s => is)
CC   1 coordinating conjunction
CD   2 cardinal number
CO   in {C...} region
DM   in {D...} region
DT   3 determiner
ET   in {E...} region
EX   4 expletive (e.g. there/EX is ...)
FW   5 foreign word
HVS  SWBD-special: HAVE-s ('s => has)
IN   6 preposition
JJ   7 adjective
JJR  8 adj./comparative
JJS  9 adj./superlative
LS   10 list item marker (e.g. we show 1/LS ... and 2/LS ...)
MD   11 modal verb
NN   12 common noun
NNP  13 proper noun
NNPS 14 proper noun, plural
NNS  15 common noun, plural
```

```
PDT  16 predeterminer (e.g. all/PDT these people...)
POS  17 possessive ending
PRP  18 personal pronoun
PRP$ 19 possessive pronoun
RB   20 adverb
RBR  21 adverb, comparative
RBS  22 adverb, superlative
RP   23 particle
SYM  24 symbol
TO   25 to (+ infinitive)
UH   26 non-lexicalized filled pause
VB   27 verb, base form
VBD  28 verb/past
VBG  29 verb/gerund
VBN  30 verb/past participle
VBP  31 verb/present, non-3rd singular
VBZ  32 verb/present, 3rd singular
WDT  33 wh-determiner
WP   34 wh-pronoun
WP$  35 possessive wh-pronoun
WRB  36 wh-adverb

NOTE: GW is not present since we create a single word if
      a GW tag is present: drug/GW testing/NN => drugtesting/NN
      XX is not present since we removed all incomplete (and XX-marked)
      words from the input
```

Appendix B

Example Annotations

B.1 Topic and Relevance Annotation

This section presents an example annotation of a topical segment from an English CALLHOME dialogue. The first part is the topic header. It contains

- a brief description of this topic's contents
- three questions about the most relevant contents (these should be answerable when reading the summary, according to the annotation guidelines)
- the turns which contain the correct answer
- the correct answer
- three multiple-choice distractor answers

After the topic header follows the corresponding section of the original transcript file with turn-ID (added), start and end times, speaker ID and finally the words themselves. Curly brackets mark various types of noises, stars typically mark filled pauses, round brackets mark areas which were not clearly audible, and ampersands mark proper names.

The annotators' mark-ups are the following types:

1. n[... n]: nucleus regions (nucleus-IUs)

2. s[... s]: satellite regions (satellite-IUs)

3. + : words with higher salience within a region: a summary just containing the '+'-marked words should still be readable and understandable

```
#TOPIC    speaker B discusses her possible plans with speaker A
#####
#Q Which date can Speaker B not make any plans for because of a possible visit?
#A The 29th.
#T 14-31
#M The 10th.
#M The 4th.
#M The 16th.
#####
#Q Which city did Speaker B speak of her visitors flying in to?
#A Boston.
#T 38
#M New York.
#M Baton-Rogue.
#M Pittsburgh.
#####
#Q Speaker B spoke of possibily visiting Speaker A while enroute to which of the following?
#A New Hampshire.
#T 42
#M South Carolina.
#M Texas.
#M Maryland.
#EOT#
     1 900.70 902.05 B: and they're going to be here
     2 902.91 903.16 B: well
     3 903.35 905.79 B: if I come out to visit you that (( )) will
                        make it easier too, I can --
     4 903.89 904.04 A: (( ))
     5 906.09 906.33 B: -- {laugh}
     6 906.87 907.80 B: see them at the same time.
     7 907.93 908.94 A: When are they going to be?
     8 908.91 911.19 B: *um, well, she goes to ((&Wilston)) &Northampton.
     9 909.26 909.81 A: (( ))
    10 911.25 911.61 A: uh-huh.
    11 911.64 911.86 B: {breath}
    12 911.74 912.17 A: {sniff}
    13 911.96 913.43 B: And they she said
```

```
14 913.58 914.36 B: *um the s[ +mom
15 914.56 916.46 B: +I +just +spoke +to +her s] a few minutes actually
16 916.55 918.69 B: sh- I suppose I was talking to her when
                    you tried to call. {breath}
17 918.86 919.46 B: *um
18 919.76 920.39 B: been a busy night.
19 920.57 921.28 B: {laugh}
20 920.59 920.75 A: m-
21 921.06 921.36 A: (( ))
22 921.62 922.10 B: {breath}
23 922.29 923.00 B: *um
24 923.73 924.73 B: I think she said sh-
25 924.85 926.44 B: n[ +they're +coming +maybe +on +the +twenty-ninth.
26 926.69 929.35 B: So and I told them if they needed any help
27 926.70 927.33 A: {lipsmack} okay
28 927.54 929.04 A:
29 930.78 931.47 B: *um
30 931.86 932.22 B: you know
31 932.31 935.45 B: I +I +kind +of I +can't +make +any +plans
                    +for +the +twenty-ninth n] until I know
32 935.23 935.62 A: {sniff}
33 935.82 936.27 A: mhm.
34 936.48 936.85 B: {breath}
35 936.88 938.45 A: Sure ((no that's fine))
36 937.00 937.15 B: But
37 937.54 939.11 B: but basically, I mean
38 939.62 942.62 B: heck it might turn out that you know n[ if
                    +they +fly in +to +&Boston I can n]
39 943.00 943.95 B: s[ +drive +them --
40 944.72 945.59 A: +To +western +&Mass. s]
41 945.44 946.71 B: -- out to western &Mass and
42 946.82 948.32 B: n[ +visit +you +on our +way +to +&New
                    +&Hampshire. n]
43 949.25 949.41 A: {lipsmack}
44 949.36 952.22 B: {laugh}
45 949.91 950.61 A: Well {breath}
46 950.91 951.38 A: Right.
47 952.27 954.75 A: I don't know. I want you for more than
                    a couple of hours. So
48 954.41 954.75 B: yeah.
49 954.88 955.41 B: No, no.
50 955.60 957.12 B: I'm talking about more than a couple of hours.
51 955.62 955.90 A: ((*um))
```

Finally, we provide (a) the summary which would be generated by extracting the nucleus- and satellite-IUs from this annotation example, and (b) the summary which would be generated using the '+'-marked words only:

```
Nucleus+satellite summary:
    B: s[ +mom
    B: +I +just +spoke +to +her s]
    B: n[ +they're +coming +maybe +on +the +twenty-ninth.
    B: So and I told them if they needed any help
    A: okay
    B: you know
    B: I +I +kind +of I +can't +make +any +plans +for +the +twenty-ninth n]
    B: n[ if +they +fly in +to +&Boston I can n]
    B: s[ +drive +them --
    A: +To +western +&Mass. s]
    B: n[ +visit +you +on our +way +to +&New +&Hampshire. n]

'+'-marked summary:
    B: s[ +mom
    B: +I +just +spoke +to +her s]
    B: n[ +they're +coming +maybe +on +the +twenty-ninth.
    B: +I +kind +of +can't +make +any +plans +for +the +twenty-ninth n]
    B: n[ +they +fly +to +&Boston n]
    B: s[ +drive +them --
    A: +To +western +&Mass. s]
    B: n[ +visit +you +on +way +to +&New +&Hampshire. n]
```

B.2 Disfluency Annotation

For disfluency annotation, the guidelines of (Meteer et al., 1995) were closely followed. We did not, however, have any {A...} sections annotated here (*asides*). The following annotations are used:

- {C...}: empty coordinating conjunctions (e.g., *and then*)
- {D...}: discourse markers (i.e., *lexicalized filled pauses* in our terminology, e.g., *you know*)
- {E...}: editing terms (within repairs, e.g., *I mean*)

- {F...}: filled pauses (non-lexicalized, e.g., *uh*)
- [... + ...]: repairs: the part before the '+' is called reparandum (to be removed), the part after the '+' repair (proper).

Further, end of sentence boundaries are marked with '/', possibly modified by a '-' (if incomplete) or a 'q' (if a question).

We now present a short example from these annotations (the same topical segment as above.)

```
900.70 902.05 B: {C and } they're going to be here /
902.91 903.16 B: {D well } --
903.35 905.79 B: -- if I come out to visit you that (( )) will make
                 it easier too, / I can --
903.89 904.04 A: (( )) /
906.09 906.33 B: -- {laugh} --
906.87 907.80 B: -- see them at the same time. /
907.93 908.94 A: When are they going to be? q/
908.91 911.19 B: {F %um, } {D well, } she goes to
                 ((&Wilston)) &Northampton. /
909.26 909.81 A: (( )) /
911.25 911.61 A: uh-huh. /
911.64 911.86 B: {breath}
911.74 912.17 A: {sniff}
911.96 913.43 B: {C And } [ they + she ] said --
913.58 914.36 B: -- {F %um } the mom -/ --
914.56 916.46 B: -- I just spoke to her a few minutes actually / --
916.55 918.69 B: -- [ sh- + ] I suppose I was talking to her when you
                 tried to call. {breath} /
918.86 919.46 B: {F %um } /
919.76 920.39 B: been a busy night. /
920.57 921.28 B: {laugh}
920.59 920.75 A: [ m- + ] -/
921.06 921.36 A: (( )) /
921.62 922.10 B: {breath}
922.29 923.00 B: {F %um } /
923.73 924.73 B: I think she said [ sh- + --
924.85 926.44 B: -- they're ] coming maybe on the twenty-ninth. / --
926.69 929.35 B: {D So } {C and } I told them if they needed any help -/
926.70 927.33 A: {lipsmack} okay /
927.54 929.04 A: [distortion]
930.78 931.47 B: {F %um } /
931.86 932.22 B: {D you know } --
```

```
932.31 935.45 B: -- [ [ I + I kind of ] + I can't ] make any plans
                 for the twenty-ninth until I know /
935.23 935.62 A: {sniff}
935.82 936.27 A: mhm. /
936.48 936.85 B: {breath}
936.88 938.45 A: Sure / ((no that's fine)) /
937.00 937.15 B: [ {C But } + --
937.54 939.11 B: -- {C but } ] basically, {E I mean } --
939.62 942.62 B: -- heck it might turn out that {D you know }
                 if they fly in to &Boston I can --
943.00 943.95 B: -- drive them --
944.72 945.59 A: To western &Mass. /
945.44 946.71 B: -- out to western &Mass / {C and } --
946.82 948.32 B: -- visit you on our way to &New &Hampshire. /
949.25 949.41 A: {lipsmack}
949.36 952.22 B: {laugh}
949.91 950.61 A: {D Well } {breath} --
950.91 951.38 A: -- Right. /
952.27 954.75 A: I don't know. / I want you for more
                 than a couple of hours. / {C So } -/
954.41 954.75 B: yeah. /
954.88 955.41 B: No, no. / --
955.60 957.12 B: -- I'm talking about more than a couple of hours. /
955.62 955.90 A: (({F %um }))
```

Appendix C

Data Format

In this appendix we explain and present the data format which DIASUMM uses throughout the pipeline architecture to keep track of the various kinds of information it acquires in the process. Most of the relevant data is kept in a table-like file format, one line for every word (token) of a dialogue (see example below).

The fields of this table have the following meaning:

1. dialogue ID
2. original turn ID
3. word ID [after tokenization]
4. speaker ID
5. start time
6. end time
7. word identity
8. ASR confidence score [1.0 if a human transcript]
9. POS tag [can be replaced by disfluency tags at later stages: FS=false start, REP=repetition]
10. sentence boundary marker (eos=sentence starts before the current word; neos=no sentence boundary before current word)

```
en_5573 1 1 b 900.70 900.876086956522 and 1.0 CO eos
en_5573 1 2 b 900.876086956522 901.110869565217 they 1.0 PRP none
en_5573 1 3 b 901.110869565217 901.286956521739 ’re 1.0 VBP none
en_5573 1 4 b 901.286956521739 901.580434782609 going 1.0 VBG none
en_5573 1 5 b 901.580434782609 901.697826086957 to 1.0 TO none
en_5573 1 6 b 901.697826086957 901.815217391304 be 1.0 VB none
en_5573 1 7 b 901.815217391304 902.05 here 1.0 RB none
en_5573 2 8 b 902.91 903.16 well 1.0 DM eos
en_5573 3 9 b 903.35 903.453829787234 if 1.0 IN none
en_5573 3 10 b 903.453829787234 903.505744680851 i 1.0 PRP none
en_5573 3 11 b 903.505744680851 903.713404255319 come 1.0 VBP none
en_5573 3 12 b 903.713404255319 903.86914893617 out 1.0 IN none
en_5573 3 13 b 903.86914893617 903.972978723404 to 1.0 TO none
en_5573 3 14 b 903.972978723404 904.232553191489 visit 1.0 VB none
en_5573 3 15 b 904.232553191489 904.38829787234 you 1.0 PRP none
en_5573 3 16 b 904.38829787234 904.595957446808 that 1.0 DT none
en_5573 3 17 b 904.595957446808 904.803617021277 will 1.0 MD none
en_5573 3 18 b 904.803617021277 905.011276595745 make 1.0 VB none
en_5573 3 19 b 905.011276595745 905.115106382979 it 1.0 PRP none
en_5573 3 20 b 905.115106382979 905.426595744681 easier 1.0 JJR none
en_5573 3 21 b 905.426595744681 905.582340425532 too 1.0 RB none
en_5573 3 22 b 905.582340425532 905.634255319149 i 1.0 PRP eos
en_5573 3 23 b 905.634255319149 905.79 can 1.0 MD none
en_5573 6 24 b 906.87 907.0095 see 1.0 DM eos
en_5573 6 25 b 907.0095 907.1955 them 1.0 PRP none
en_5573 6 26 b 907.1955 907.2885 at 1.0 IN none
en_5573 6 27 b 907.2885 907.428 the 1.0 DT none
en_5573 6 28 b 907.428 907.614 same 1.0 JJ none
en_5573 6 29 b 907.614 907.8 time 1.0 NN none
en_5573 7 30 a 907.93 908.132 when 1.0 WRB eos
en_5573 7 31 a 908.132 908.2835 are 1.0 VBP none
en_5573 7 32 a 908.2835 908.4855 they 1.0 PRP none
en_5573 7 33 a 908.4855 908.738 going 1.0 VBG none
en_5573 7 34 a 908.738 908.839 to 1.0 TO none
en_5573 7 35 a 908.839 908.94 be 1.0 VB none
en_5573 8 36 b 908.91 909.048181818182 um 1.0 UH eos
en_5573 8 37 b 909.048181818182 909.324545454545 well 1.0 DM eos
en_5573 8 38 b 909.324545454545 909.531818181818 she 1.0 PRP none
en_5573 8 39 b 909.531818181818 909.808181818182 goes 1.0 VBZ none
en_5573 8 40 b 909.808181818182 909.946363636364 to 1.0 IN none
en_5573 8 41 b 909.946363636364 910.43 wilston 1.0 NNP none
en_5573 8 42 b 910.43 911.19 northampton 1.0 NNP none
```

```
en_5573 10 43 a 911.25 911.61 uh-huh 1.0 UH eos
en_5573 13 44 b 911.96 912.275 and 1.0 CO eos
en_5573 13 45 b 912.275 912.695 they 1.0 PRP none
en_5573 13 46 b 912.695 913.01 she 1.0 PRP none
en_5573 13 47 b 913.01 913.43 said 1.0 VBD none
en_5573 14 48 b 913.58 913.775 um 1.0 UH eos
en_5573 14 49 b 913.775 914.0675 the 1.0 DT none
en_5573 14 50 b 914.0675 914.36 mom 1.0 NN none
en_5573 15 51 b 914.56 914.615882352941 i 1.0 PRP eos
en_5573 15 52 b 914.615882352941 914.839411764706 just 1.0 RB none
en_5573 15 53 b 914.839411764706 915.118823529412 spoke 1.0 VBD none
en_5573 15 54 b 915.118823529412 915.230588235294 to 1.0 IN none
en_5573 15 55 b 915.230588235294 915.398235294118 her 1.0 PRP$ none
en_5573 15 56 b 915.398235294118 915.454117647059 a 1.0 DT none
en_5573 15 57 b 915.454117647059 915.621764705882 few 1.0 JJ none
en_5573 15 58 b 915.621764705882 916.01294117647 minutes 1.0 NNS none
en_5573 15 59 b 916.01294117647 916.46 actually 1.0 RB none
en_5573 16 60 b 916.55 916.600952380952 i 1.0 PRP eos
en_5573 16 61 b 916.600952380952 916.957619047619 suppose 1.0 VBP none
en_5573 16 62 b 916.957619047619 917.008571428571 i 1.0 PRP eos
en_5573 16 63 b 917.008571428571 917.161428571428 was 1.0 VBD none
en_5573 16 64 b 917.161428571428 917.518095238095 talking 1.0 VBG none
en_5573 16 65 b 917.518095238095 917.62 to 1.0 IN none
en_5573 16 66 b 917.62 917.772857142857 her 1.0 PRP$ none
en_5573 16 67 b 917.772857142857 917.976666666667 when 1.0 WRB none
en_5573 16 68 b 917.976666666667 918.129523809524 you 1.0 PRP eos
en_5573 16 69 b 918.129523809524 918.384285714286 tried 1.0 VBD none
en_5573 16 70 b 918.384285714286 918.48619047619 to 1.0 TO none
en_5573 16 71 b 918.48619047619 918.69 call 1.0 VB none
en_5573 17 72 b 918.86 919.46 um 1.0 UH eos
en_5573 18 73 b 919.76 919.94 been 1.0 VBN eos
en_5573 18 74 b 919.94 919.985 a 1.0 DT none
en_5573 18 75 b 919.985 920.165 busy 1.0 JJ none
en_5573 18 76 b 920.165 920.39 night 1.0 NN none
```

Appendix D

Instructions and Examples for User Studies

In this appendix, we provide the instructions and examples for the two user studies performed in our thesis: (a) the Q-A user study, focusing on informativeness and coherence changes due to the Q-A detection component; and (b) the user study focusing on information content of different types of summaries.

D.1 Question-Answer User Study

D.1.1 Instructions

Instructions: You are looking at a series of 11 tasks, each of which contains three versions of an *extract of a conversation*. Each speaker segment is preceded by a number (indicating its position in the original text) and the speaker label (name). Usually, there is one speaker segment per text which is "cut off" and marked with a double **–** at its end.

This is what we ask you to do now: Enter your name, email, and the time you start this experiment on the front sheet. Then proceed as follows in the order of the 11 tasks in the booklet (and please complete the experiment in one go, without taking breaks):

1. Read each version of the current text

2. Evaluate each version in terms of *informativeness* and *fluency*. The scales range from 1 to 5, with 1 meaning very low informativeness/fluency and 5 meaning "very high". Informativeness measures how much information the text contains ("dense" vs. "sparse" text), fluency, how easy it is to read and how coherent it is.

3. Since the versions of each text usually differ only very little, you then should focus on even the subtle differences to *rank* the three versions in the 2 dimensions of informativeness and fluency. In rare cases, when you are sure that 2 versions are absolutely identical, you are allowed to indicate equality with a = in the ranking list. (e.g. A=C B).

4. Note that the rankings are more essential than the raw values of informativeness and fluency. These raw values do not have to be consistent over different texts (since they are hard to compare), but **must** be consistent across the three versions of one particular text.

You see an example annotation below.

When you are done, please mark the end time on the front sheet. Then return the booklet to me and you will receive your compensation.

THANK YOU for your participation!

Example annotation:								
Version A	informativeness: LOW	1	2	3	(4)	5	HIGH
Version B	informativeness: LOW	1	2	(3)	4	5	HIGH
Version C	informativeness: LOW	1	2	3	4	(5)	HIGH
Version A	fluency: LOW	1	(2)	3	4	5	HIGH
Version B	fluency: LOW	1	2	3	(4)	5	HIGH
Version C	fluency: LOW	1	2	(3)	4	5	HIGH
Ranking of A,B,C:	informativeness: TOP			C A B			BOTTOM
Ranking of A,B,C:	fluency: TOP			B C A			BOTTOM

D.1.2 Example

```
[No Q-A detection:]
  46         b: I'm going on the february eighth
  55        a1: As a possible retirement area
  61        a1: Look it over from that perspective to whatever extent
                 you can
  73         b: It's in the --
  81         b: It's like what they have in brazil where the poor
                 people dress up like the rich

[DiaSumm QA-detection:]
  45         a: When are you going --
  46         b: I'm going on the february eighth
  55        a1: As a possible retirement area
  61        a1: Look it over from that perspective to whatever extent
                 you can
  81         b: It's like what they have in brazil where the poor
                 people dress up like the rich

[Optimal Q-A detection:]
  61        a1: Look it over from that perspective to whatever extent
                 --
  76         a: Now what is carnival
  77         a: Does that have to do with easter or not
  78         a: No
  79         a: It's not
  80         b: Mm
  81         b: It's like what they have in brazil where the poor
                 people dress up like the rich
```

D.2 Multiple-Choice User Study

D.2.1 Instructions

In this experiment, you will be given 34 texts to read. For each text, there are 3 multiple choice questions for you to answer.

You should try to be as fast and accurate as possible. There may be cases where there is no answer to a question in a text. BUT: Even if you don't think that you can answer a question, try to make your **best-guess choice**, as you would do on a test.

You may take breaks, but ONLY AFTER you have submitted a set of three answers and BEFORE you load a new text.

Specific Information:

Each text is an extract from a conversation transcript between several people. Every line starts with a number (think of it as a sentence number in the dialog), followed by the speaker-ID, and then the text itself. (Often, there are "gaps" between line numbers to indicate that some part of the conversation is missing here.) The text is sometimes "shortened", e.g., missing many words from the original and thus does not always have a proper sentence structure.

If a text is long and fills the entire text window, you should use the scrollbar to be able to read the whole text (most texts will however be shorter and not require this.)

Procedure Information:

The steps in the experiment's user interface are as follows:

1. Click "Next Text" to load the next text
2. Read the text (possibly using the scrollbar); and then, for every question...:
   (a) Read the question
   (b) Select your answer choice
   (c) Click "Submit Answer"
3. Goto 1. [only before this step you may take a break]

Note: After the last text, you will be notified about the end of the experiment on the screen; please notify me that you are done and you will receive your reimbursement subsequently.

Thank you very much for your participation!!

D.2.2 Example

The user interface screen is divided into two main sections: the upper part contains a summary (LEAD, MMR, NPTELE, DIASUMM, or Gold), and the lower part contains a question with 4 answer choices — the user clicks on the choice he/she feels is the most likely one given the summary above. After submitting the first answer, two more questions are presented, and then the user selects the next summary to be presented.

We present the five different summary versions for a topical segment of the English CALLHOME corpus here, along with the questions, answers (as the user sees them) and the correct answers being marked.

```
Example summaries:

Gold:
     5          a :  I'm with rachel constantly so it's funny
                     that I don't see her more actually
    19          a :  I got this offer to go to madrid I was like
    20          a :  Oh i'd much rather go visit them in
    22          a :  Spain
    26          a :  I think I might go thanksgiving that week

LEAD:
     1          a :  Um I know cynthia through rach
     2          b :  Oh so she yeah so she is
     3          a :  And so it's kind of like a roundabout thing
                     and I we don't talk that much I see her
                     occasionally we all go out
     4          b :  Oh yeah
     5          a :  Um but not all that

MMR:
     3          a :  And so it's kind of like a roundabout
                     thing and I we don't talk that much I see
                     her occasionally we all go out
     5          a :  Um but not all that
    37          a :  I guess you lived there long enough to get
                     it all out of your system

DiaSumm:
     1          a :  I know cynthia through rach
     6          a :  It's kind of like a roundabout thing
     7          a :  I we don't talk that much I see her
                     occasionally we all go out
    10          a :  Not all that much I'm with rachel constantly
```

```
    54          a :  I guess you lived there long enough to get

NPtele:
     1          a :  I know cynthia through rach ...
     6          a :  It's ...  like a roundabout thing
     7          a :  I we don't talk ...  I see ...  we ...  go
    10          a :  I'm with rachel ...
    36          a :  I ...  rather go visit them in spain ...
                     I can see all the
    54          a :  I guess you lived ...  to get it ...  of your system
```

Questions and Answers:

```
Q1   Who is speaker A with constantly?
1    &Cynthia
1    Herself.
1    No one.
1*   &Rachel

Q2   When is Speaker A going to visit Spain for the wedding?
2    During spring break.
2    During Easter.
2*   During Thanksgiving.
2    During Christmas.

Q3   Where would the woman rather visit her friends?
3    Madrid.
3    At the wedding.
3*   In Spain.
3    When they go out.
```