A Skip-Chain Conditional Random Field for Ranking Meeting Utterances by Importance*

Michel Galley
Columbia University
Department of Computer Science
New York, NY 10027, USA
galley@cs.columbia.edu

Abstract

We describe a probabilistic approach to content selection for meeting summarization. We use *skip-chain Conditional Random Fields* (CRF) to model non-local pragmatic dependencies between paired utterances such as QUESTION-ANSWER that typically appear together in summaries, and show that these models outperform linear-chain CRF and Bayesian models in the task. We also discuss different approaches for ranking all utterances in a sequence using CRFs. Our best performing system achieves 91.3% of human performance when evaluated with the Pyramid evaluation metric, which represents a 3.9% absolute increase compared to our most competitive non-sequential classifier.

1 Introduction

Summarization of meetings faces many challenges not found in texts, i.e., high word error rates, absence of punctuation, and sometimes lack of grammaticality and coherent ordering. On the other hand, meetings present a rich source of structural and pragmatic information that makes summarization of multi-party speech quite unique. In particular, our analyses of patterns in the verbal exchange between participants found that *adjacency pairs* (AP), a concept drawn from the conversational analysis literature (Schegloff and Sacks, 1973), have particular relevance to summarization. APs are pairs of utterances such as QUESTION-ANSWER or OFFER-ACCEPT, in which the second utterance is said to be conditionally relevant on the first. We show that there is a strong correlation between the two elements of an AP in summarization, and that one is unlikely to be included if the other element is not present in the summary.

Most current statistical sequence models in natural language processing, such as hidden Markov models (HMMs) (Rabiner, 1989), are linear chains that only encode local dependencies between utterances to be labeled. In multi-party speech, the two elements of an AP are generally arbitrarily distant, and such models can only poorly account for dependencies underlying APs in summarization. We use instead skip-chain sequence models (Sutton and McCallum, 2004), which allow us to explicitly model dependencies between distant utterances, and turn out to be particularly effective in the summarization task.

In this paper, we compare two types of network structures—linear-chain and skip-chain—and two types of network semantics—Bayesian Networks (BN) and Conditional Random Fields (CRF). We discuss the problem of computing the class posterior probability of each utterance in a sequence in order to extract the N most probable ones, and show that the cost assigned by a CRF to each utterance needs to be locally normalized in order to outperform BNs. After analyzing the predictive power of a large set of durational, acoustical, lexical, structural, and information retrieval features, we perform feature selection to have a competitive set of predictors to test the different models. Empirical evaluations using two standard summarization metrics—the Pyramid method (Nenkova and Passonneau, 2004b) and ROUGE (Lin, 2004)—show that the best performing system is a CRF incorporating both order-2 Markov dependencies and skip-chain dependencies, which achieves 91.3% of human performance in Pyramid score, and outperforms our best-performing non-sequential model by 3.9%.

2 Corpus

The work presented here was applied to the ICSI Meeting Corpus (Janin et al., 2003), a corpus of "naturally-occurring" meetings, i.e. meetings that would have taken place anyway. Their style is quite informal, and topics are primarily concerned with speech, natural language, artificial

*This material is based on research supported by the U.S. National Science Foundation (NSF) under Grant No. IIS-0121396. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the NSF.

intelligence, and networking research. The corpus contains 75 meetings, which are 60 minutes long on average, and involve a number of participants ranging from 3 to 10 (6 on average). The total number of unique speakers is 60, including 26 non-native English speakers. Experiments in this paper are based either on human orthographic transcriptions or automatic speech recognition output, which were available for all meetings. For automatic recognition, we used the ICSI-SRI-UW speech recognition system, (Mirghafori et al., 2004), a state-of-the-art conversational telephone speech (CTS) recognizer whose language and acoustic models were adapted to the meeting domain. It achieves 34.8% WER on the ICSI corpus, which is indicative of the difficulty involved in processing meetings automatically.

We also used additional annotation that has been developed to support higher-level analyses of meeting structure, in particular the ICSI Meeting Recorder Dialog act (MRDA) corpus (Shriberg et al., 2004). Dialog act (DA) labels describe the pragmatic function of utterances, e.g. a STATEMENT or a BACKCHANNEL. This auxiliary corpus consists of over 180,000 human-annotated dialog act labels ($\kappa = .8$), for which so-called *adjacency pair* (AP) relations (e.g., APOLOGY-DOWNPLAY) were also labeled. This latter annotation was used to train an AP classifier that is instrumental in automatically determining the structure of our sequence models, as we will describe in Section 4. Note that, in the case of three or more speakers, *adjacency pair* is admittedly an unfortunate term, since labeled APs are generally not adjacent (e.g., see Table 1), but we will nevertheless use the same terminology to enforce consistency with previous work.

To train and evaluate our summarizer, we used a corpus of extractive summaries produced at the University of Edinburgh (Murray et al., 2005). For each of the 75 meetings, human judges were asked to select transcription utterances segmented by DA to include in summaries, resulting in an average compression factor of about 92.7% (though no strict limit was imposed). Inter-labeler agreement was measured using six meetings that were summarized by multiple coders (average $\kappa = .323$). While this level of agreement is admittedly quite low, this situation is not uncommon to summarization, since there may arguably be many good summaries for a given document; a main challenge lies in using evaluation schemes that properly accounts for this diversity.

3 Content selection

State sequence Markov models such as hidden Markov models (Rabiner, 1989) have been highly successful in many speech and natural language processing applications, including summarization. Following an intuition that the probability of a given sentence may be locally conditioned on the previous one, Conroy (2004) built a HMM-based summarizer that consistently ranked among the top systems in recent Document Understanding Conference (DUC) evaluations.

Inter-sentential influences become more complex in the case of dialogues or correspondences, especially when they involve multiple parties. In the case summarization of conversational speech, Zechner (2002) found, for instance, that a simple technique consisting of linking together questions and answers in conversational speech summaries—and thus preventing the selection of orphan questions or answers—significantly improved their readability according to various human summary evaluations. In email summarization (Rambow et al., 2004), Shrestha and McKeown (2004) obtained good performance in automatic detection of questions and answers, which can help produce summaries that highlight or focus on the question and answer exchange. In a combined chat and email summarization task, a technique (Zhou and Hovy, 2005) consisting of identifying APs and appending any relevant responses to topic initiating messages was instrumental in outperforming two competitive summarization baselines.

The need to model pragmatic influences, such as between a question and an answer is also prevalent in meeting summarization. In fact, question-answer pairs are not the only discourse relations that we need to preserve in order to create coherent summaries, and as we will see, most instances of AP would need to be preserved together, either inside or outside the summary. Table 1 displays an AP construction with one question (A part) and three respondents (B parts). This example illustrates that the number of turns between constituents of APs is variable and thus difficult to model with standard sequence models. This example also illustrates some of the predictors investigated in this paper. First, many speakers respond to A's utterance, which is generally a strong

Time	Speaker	AP	Transcript
1480.85-1493.91	1	A	are - are those d- delays adjustable? see a lot of people who actually build stuff with human computer interfaces understand that delay, and - and so when you - by the time you click it it'll be right on because it'll go back in time to put the -
1489.71-1489.94	2		*yeah.*
1493.95-1495.41	3	B	*yeah, uh, not in this case.*
1494.31-1495.83	2	B	*it could do that, couldn't it.*
1495.1-1497.07	4	B	we could program that pretty easily , couldn't we?

Table 1: Snippet of a meeting displaying an AP construction, where a question (A) initiates three responses (B). Sentences in *italic* are not present in the reference summary.

indicator that the A utterance should be included. Secondly, while APs are generally characterized in terms of pre-defined dialog acts, such as OFFER-ACCEPT, we found that the type of dialog act has much less importance than the existence of the AP connection itself. Since DAs seem to matter less than adjacency pairs, the aim will be to build techniques to automatically identify such relations and exploit them in utterance selection.

In the current work, we use skip-chain sequence models (Sutton and McCallum, 2004) to represent dependencies between both contiguous utterances and paired utterances appearing in the same AP constructions. The graphical representations of skip-chain models, such as the skip-chain CRF represented in Figure 1, are composed of two types of edges: linear-chain and skip-chain edges. The latter edges model AP links, which we represent as a set of (s, d) index pairs (note that no AP may share the same second element d).

The intuition that the summarization labels (-1 or 1) are highly correlated with APs is confirmed in Table 2. While contiguous labels y_{t-1} and y_t seem to seldom influence each other, the correlation between AP elements y_s and y_d is particularly strong, and they have a tendency to be either both included or both excluded. Note that the second table is not symmetric, because as seen in Table 1, the data allows an A part to be linked to multiple B parts, but not vice-versa. While counts in Table 2 reflect human labels, we only use automatically predicted (s, d) pairs in the experiments of the remaining part of this paper. To find these pairs automatically, we trained a non-sequential log-linear model that achieves a .902 accuracy (Galley et al., 2004).

4 Skip-Chain Sequence Models

In this paper, we investigate conditional models for paired sequences of observations and labels. In the case of utterance selection, the observation sequence $\mathbf{x} = \mathbf{y}_{1:T} = (x_1, \ldots, x_T)$ represents local

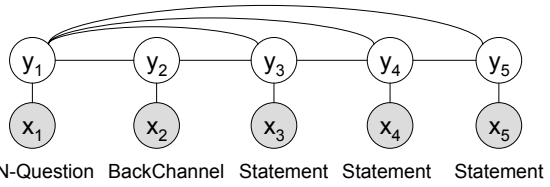

Figure 1: A skip-chain CRF with pragmatic-level links.

Linear-chain edges	$y_t = 1$	$y_t = -1$
$y_{t-1} = 1$	529	7742
$y_{t-1} = -1$	7742	116040
Skip-chain edges	$y_d = 1$	$y_d = -1$
$y_s = 1$	6792	2191
$y_s = -1$	1479	121591

Table 2: Contingency tables: while the correlation between adjacent labels y_{t-1} and y_t is not significant ($\chi^2 = 2.3$, $p > .05$), empirical evidence clearly shows that y_s and y_d influence each other ($\chi^2 = 78948$, $p < .001$).

summarization predictors (see Section 6), and the binary sequence $\mathbf{y} = \mathbf{y}_{1:T} = (y_1, \ldots, y_T)$ (where $y_t \in \{-1, 1\}$) determines which utterances to include in the summary. In a discriminative framework, we concentrate our modeling effort on estimating $p(\mathbf{y}|\mathbf{x})$ from data, and do not explicitly model the prior probability $p(\mathbf{x})$, since $\mathbf{x}$ is fixed during testing anyway.

Many probabilistic approaches to modeling sequences have relied on directed graphical models, also known as Bayesian networks (BN),[1] in particular hidden Markov models (Rabiner, 1989) and conditional Markov models (McCallum et al., 2000). However, prominent recent approaches have focused on undirected graphical models, in particular conditional random fields (CRF) (Lafferty et al., 2001), and provided state-of-the-art performance in many natural language processing tasks. In our work, we will provide empirical results for state sequence models of both semantics,

[1] In the existing literature, sequence models that satisfy the Markovian condition—i.e., the state of the system at time t depend only on its immediate past $t-k : t-1$, typically just $t-1$— are generally termed dynamic Bayesian networks (DBN). Since the particular models under investigation, i.e. skip-chain models, do not have this property, we will simply refer to them as Bayesian networks.

and we will now describe skip-chain models for both BNs and CRFs.

In a BN, the probability of the sequence $\mathbf{y}$ factorizes as a product of probabilities of local predictions y_t conditioned on their parents $\pi(y_t)$ (Equation 1). In a CRF, the probability of the sequence $\mathbf{y}$ factorizes according to a set of clique potentials $\{\Phi_c\}_{c \in C}$, where C is represents the cliques of the underlying graphical model (Equation 2).

$$p_{\text{BN}}(\mathbf{y}|\mathbf{x}) = \prod_{i=1}^{T} p_{\text{BN}}(y_t|\mathbf{x}, \pi(y_t)) \quad (1)$$

$$p_{\text{CRF}}(\mathbf{y}|\mathbf{x}) \propto \prod_{c \in C} \Phi_c(\mathbf{x}_c, \mathbf{y}_c) \quad (2)$$

We parameterize these BNs and CRFs as log-linear models, and factorize both BN's local prediction probabilities and CRF's clique potentials using two types of feature functions. *Linear-chain* feature functions $f_j(\mathbf{y}_{t-k:t}, \mathbf{x}, t)$ represent local dependencies that are consistent with an order-k Markov assumption. For instance, such a function could be predicate that returns 1 if and only if $y_{t-1} = 1$, $y_t = -1$, and (x_{t-1}, x_t) indicates that both utterances are produced by the same speaker. Given a set of skip edges $\mathcal{S} = \{(s_t, t)\}$ specifying source and destination indices, *skip-chain* feature functions $g_j(y_{s_t}, y_t, \mathbf{x}, s_t, t)$ exploit dependencies between variables that are arbitrarily distant in the chain. For instance, the finding that OFFER-REJECT pairs are often linked in summaries might be a encoded as a skip-chain feature predicate equal to 1 if and only if $y_{s_t} = 1$, $y_t = 1$, and the first word of the t-th utterance is "no".

Log-linear models for skip-chain sequence models are defined in terms of weights $\{\lambda_k\}$ and $\{\mu_k\}$, one for each feature function. In the case of BNs, we write:

$$\log p_{\text{BN}}(y_t|\mathbf{x}, \pi(y_t)) \propto \sum_{j=1}^{J} \lambda_j f_j(\mathbf{x}, \mathbf{y}_{t-k:t}, t) + \sum_{j=1}^{J'} \mu_j g_j(\mathbf{x}, y_{s_t}, y_t, s_t, t)$$

As seen in Figure 1, the particular structure of skip-chain CRFs reduces the set of cliques to (y_{t-1}, y_t) adjacency edges and (y_{s_t}, y_t) skip edges, resulting in only two potential functions:

$$\log \Phi_{\text{LIN}}(\mathbf{x}, \mathbf{y}_{t-k:t}, t) = \sum_{j=1}^{J} \lambda_j f_j(\mathbf{x}, \mathbf{y}_{t-k:t}, t)$$

$$\log \Phi_{\text{SKIP}}(\mathbf{x}, y_{s_t}, y_t, t) = \sum_{j=1}^{J'} \mu_j g_j(\mathbf{x}, y_{s_t}, y_t, s_t, t)$$

4.1 Inference and Parameter Estimation

Our CRF and BN models were designed using MALLET (McCallum, 2002), which provides tools for training log-linear models with L-BFGS optimization techniques and maximize the log-likelihood of our training data $\mathcal{D} = (\mathbf{x}^{(i)}, \mathbf{y}^{(i)})_{i=1}^{N}$. It implements the Viterbi algorithm for decoding with linear-chain BNs and CRFs of arbitrary order.

To account for skip-edges, previous work used approximate probabilistic inference techniques, including TRP (Sutton and McCallum, 2004) and Gibbs sampling (Finkel et al., 2005). We used instead a technique inspired by (Sha and Pereira, 2003), in which multiple state dependencies, such as an order-2 Markov model, are encoded using auxiliary tags. For instance, an order-2 Markov model is parameterized using state triples $y_{t-2:t}$, and each possible triple is converted to a label $z_t = y_{t-2:t}$. Using these auxiliary labels only, we can then use the standard forward-backward algorithm for probabilistic inference in linear-chain CRFs, and Viterbi decoding in linear-chain CRFs and BNs. The only requirement is to ensure that a transition between z_t and z_{t+1} is forbidden if the sub-states $y_{t-1:t}$ common two both states differ, and is assigned an infinite cost. This approach can be extended to the case of skip-chain transitions. For instance, an order-1 Markov model with skip-edges can be constructed using $z_t = (y_{s_t}, y_{t-1}, y_t)$ triples, where the first element y_{s_t} represents the label at the source of the skip-edge. Similarly to the case of order-2 Markov models, we need to ensure that only valid sequences of labels are considered. Even though this approach is not exact, it still provides competitive performance as we will see in Section 8. In future work, we plan to explore other probabilistic inference techniques.

5 Ranking Utterances by Importance

As we will see in Section 8, using the actual $\{-1, 1\}$ label predictions of our BNs and CRFs leads to significantly sub-optimal results, which might be explained by the following reasons. First, our models are optimized to maximize the conditional log-likelihood of the training data, a measure that does not correlate well with utility measures generally used in retrieval oriented tasks such as summarization, especially when faced with a significant class imbalance (only 7.8% of

reference instances are positive).[2] Second, the MAP decision rule doesn't give us the freedom to select an arbitrary number of sentences, and in the case of the discriminatively trained models, hurts performance with recall-oriented metrics common to summarization.

A solution is to compute the posterior probability of each local prediction y_t, and extract the N most probable summary sentences $(y_{r_1}, \ldots, y_{r_k})$, where N is generally not explicitly given, but depends on a length limit expressed in number of words, and the length of the best scoring utterances.

In contrast to CRFs, BNs assign probability distributions over entire sequences by estimating the probability of each individual instance y_t in the sequence (Equation 1); thus, it initially seems that they are better suited for the task; yet, we would like to exploit CRF's effective ability to model sequential data, since they outperform directed models in many tasks described in the literature.

A first approach is to rank utterances according to the costs of predictions $y_t = 1$ that can be read from the Viterbi table. While these costs are well-formed (negative log) probabilities in the case of BNs, they cannot be interpreted as such in the case of CRFs, and turn out to produce poor results with CRFs. Since BNs and CRFs are here parameterized as log-linear models and rely on the same set of feature functions, a second approach is to use CRF-trained model parameters to build a BN classifier that assigns a probability to each y_t. Given an observation sequence $\mathbf{x}$ and a CRF predicted sequence $\hat{\mathbf{y}}$, we assign the following probability to each assignment $y_t = 1$:

$$\log p_{\text{local-CRF}}(y_t = 1 | \mathbf{x}, \hat{\mathbf{y}}_{1:t-1}) \propto$$
$$\sum_{j=1}^{J} \lambda_j f_j(\mathbf{x}, \hat{\mathbf{y}}_{t-k:t-1}, 1, t) + \sum_{j=1}^{J'} \mu_j g_j(\mathbf{x}, \hat{y}_{s_t}, 1, s_t, t)$$

Note that this local normalization step is only performed to get more sensible costs for each individual prediction, and that we do not change CRFs decoding output, even in the cases where any label $(-\hat{y}_t)$ has higher probability according to $p_{\text{local-CRF}}$.

6 Features for extractive summarization

We started our analyses with a large collection of features found to be good predictors in either speech (Inoue et al., 2004; Maskey and Hirschberg, 2005; Murray et al., 2005) or text summarization (Mani and Maybury, 1999). Our goal is to build a very competitive feature set that capitalizes on recent advances in summarization of both genres. Table 3 lists some important features.

There is strong evidence that lexical cues such as "significant" and "great" are strong predictors in many summarization tasks (Edmundson, 1968). Such cues are admittedly quite genre specific, so we did not want to commit ourselves to any specific list, which may not carry over well to our specific speech domain, and we automatically selected a list of n-grams ($n \leq 3$) using cross-validation on the training data. More specifically, we computed the mutual information of each n-gram with the class variable, and selected for each n the 200 best scoring n-grams. Other lexical features include: the number of digits, which is help-

[2] In our particular case, discriminative training leads to high-precision low-recall classifiers whose high accuracies in the range of .92 to .94 are of course quite misleading.

Lexical features:
- n-grams ($n \leq 3$)
- number of words
- number of digits
- number of consecutive repeats

Information retrieval features:
- max/sum/mean frequency of all terms in u_t
- max/sum/mean idf score
- max/sum/mean $tf \cdot idf$ score
- cosine similarity between word vector of u_t with centroid of of the meeting
- scores of LSA with 5, 10, 50, 100, 200, 300 concepts

Acoustic features:
- seconds of silence before/during/after the turn
- speech rate
- min/max/mean/median/stddev/onset/outset f0 of utterance t, and of first and last word
- min/max/mean/stddev energy
- .05, .25, .5, .75, .95 quantiles of f0 and energy
- pitch range
- f0 mean absolute slope

Durational and structural features:
- duration of the previous/current/next utterance
- relative position within meeting (i.e., index t)
- relative position within speaker turn
- large number of structural predicates, i.e. "is the previous utterance of the same speaker?"
- number of APs initiated in y_t

Discourse features:
- lexical cohesion score (for topic shifts) (Hearst, 1994)
- first and second word of utterance, if in cue word list
- number of pronouns
- number of fillers and fluency devices (e.g., "uh", "um")
- number of backchannel and acknowledgment tokens (e.g., "uh-huh", "ok", "right")

Table 3: Features for extractive summarization. Unless otherwise mentioned, we refer to features of utterance t whose label y_t we are trying to predict.

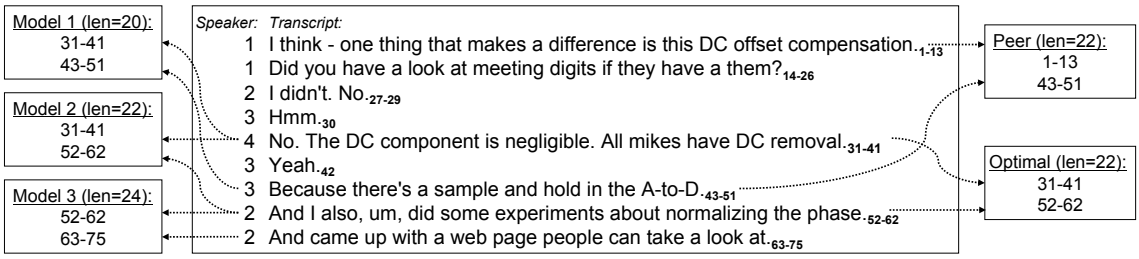

Figure 2: Model , peer, and "optimal" summaries are all extracts taken from the same transcription.

ful for identifying those sections of the meetings where participants collect data by recording digits; the number of repeats, which is meant to identify the kind of hesitations and disfluencies that negatively correlates with what is included in the summary.

The information retrieval feature set contains many features that are generally found helpful in summarization, in particular $tf \cdot idf$ and scores derived from centroid methods. In particular, we used the latent semantic analysis (LSA) feature discussed in (Murray et al., 2005), which attempts to determine sentence importance through singular value decomposition, and whose resulting singular values and singular vectors can be exploited to associate each utterance a degree of relevance to one of the top-n concepts of the meetings (where n represents the number of dimensions in the LSA). We used the same scoring mechanism as (Murray et al., 2005), though we extracted features for many different n values.

Acoustic features extracted with Praat (Boersma and Weenink, 2006) were normalized by channel and speaker, and many raw features such as f0 and mean were extracted. Structural features incorporate many that can be extracted from the sequence model before decoding, e.g., the duration that separates the two elements of an AP. Finally, discourse features represent predictors that may substitute to DA labels. While DA tagging is not directly our concern, it is presumably helpful to capitalize on discourse characteristics of utterances involved in adjacency pairs, since different types of dialog acts may be unequally likely to appear in a summary.

7 Evaluation

Evaluating summarization is a difficult problem and there is no broad consensus on how to best perform this task. Two metrics have become quite popular in the multi-document summarization community, namely the Pyramid method (Nenkova and Passonneau, 2004b) and ROUGE (Lin, 2004). In the case of single document extractive summarization, it is not clear what is the most suitable technique.

Pyramid and ROUGE are techniques looking for content units repeated in different model summaries, i.e., summary content units (SCUs) such as clauses and noun phrases for the Pyramid method, and n-grams for ROUGE. The underlying hypothesis is that different model sentences, clauses, or phrases may convey the same meaning, which is a reasonable assumption when dealing with reference summaries produced by different authors, since it is quite unlikely that any two abstractors would use the exact same words to convey the same idea.

Our situation is however quite different, since all model summaries of a given document are utterance extracts of that same document, as this can been seen in the excerpt of Figure 2. In our own annotation of three meetings with SCUs defined as in (Nenkova and Passonneau, 2004a), we found that repetitions and reformulation of the same information are particularly infrequent, and that textual units that express the same content among model summaries are generally originating from the same document sentence (e.g., in the figure, the first sentence in model 1 and 2 emanate from the same document sentence). Really short SCUs (e.g., base noun phrases) sometimes appeared in different locations of a meeting, but we think it is problematic to assume that connections between such short units is indicative of any similarity of sentential meaning: the contexts are different, and words may be uttered by different speakers, which may lead to unrelated or conflicting pragmatic forces. For instance, an SCU realized as "DC offset" and "DC component" appears in two different sentences in the figure, i.e. those identified as 1-13 and 31-41. However, the two sentences have contradictory meanings, and thus it would be

unfortunate to increase the score a peer summary containing the former sentence because the latter is included in some model summaries.

For all these reasons, we believe that summarization evaluation in our case should rely on the following restrictive matching: two summary units should be considered equivalent if and only if they are extracted from the *same location* in the original document (e.g., the "DC" appearing in models 1 and 2 is not the same as the "DC" in the peer summary, since they are extracted from different sentences). This constraint on the matching is reflected in our Pyramid evaluation, and we define an SCU as a word and its document position, which lets us distinguish ("DC",11) from ("DC",33). While this restriction on SCUs forces us to disregard scarcely occurring paraphrases and repetitions of the same information, it provides the benefit of automated evaluation.

Once all SCUs have been identified, the Pyramid method is applied as in (Nenkova and Passonneau, 2004b): we compute a score $\mathcal{D}$ by adding for each SCU present in the summary a score equal to the number of model summaries in which that SCU appears. The Pyramid score $\mathcal{P}$ is computed by dividing $\mathcal{D}$ by the maximum $\mathcal{D}^*$ value that is obtainable given the constraint on length. For instance, the peer summary in the figure gets a score $\mathcal{D} = 9$ (since the 9 SCUs in range 43-51 occur in one model), and the maximum obtainable score is $\mathcal{D}^* = 44$ (all SCUs of the optimal summary appear in exactly two model summaries), hence the peer summary's score is $\mathcal{P} = .204$.

While our evaluation scheme is similar to comparing the binary predictions of model and peer summaries—each prediction determining whether a given transcription word is included or not—and averaging precision scores over all peer-model pairs, the Pyramid evaluation differs on an important point, which makes us prefer and use the Pyramid evaluation method: the maximum possible Pyramid score is always guaranteed to be 1, but average precision scores can become arbitrarily low as the consensus between summary annotators decreases. For instance, the average precision score of the optimal summary in the figure is $PR = \frac{2}{3}$.[3] In the case of the six test meetings, which all have either 3 or 4 model summaries, the maximum possible average precision is .6405.

[3]Precision scores of the optimal summary compared against the the three model summaries are .5, 1, and .5, respectively, and hence average $\frac{2}{3}$. We can show that $\mathcal{P} = PR/PR^*$, where PR^* is the average precision of the optimal summary. Lack of space prevent us from providing a proof, so we will just show that the equality holds in our example: since the peer summary's precision scores against the three model summaries are respectively $\frac{9}{22}$, 0, and 0, we have $PR/PR^* = (\frac{9}{66})/(\frac{2}{3}) = \frac{9}{44} = \mathcal{P}$.

	FEATURES	$F_{\beta=1}$
1	utterance duration	.246
2	100-dimension LSA	.268
3	duration of utterance $t-1$	.275
4	time between utterances s and $d=t$	.281
5	IDF mean	.284
6	meeting position	.286
7	number of APs initiated in t	.288
8	duration of utterance $t+1$	.288
9	number of fillers	.289
10	.25-quantile of energy	.290
11	number of lexical repeats	.292
12	lexical cohesion score	.294
13	f0 mean of last word of utterance t	.294
14	LSA 50 dimensions	.295
15	utterances (t,$t+1$) by same speaker	.298
16	speech rate	.302
17	"is that"	.303
18	"for the"	.303
19	(u_{t-1},u_t) by same speaker	.305
20	"to try"	.305
21	"meetings"	.305
22	utterance starts with "and"	.306
23	"we have"	.306
24	"new"	.307
25	utterances starts with "what"	.307

Table 4: Forward feature selection.

8 Experiments

We follow (Murray et al., 2005) in using the same six meetings as test data, since each of these meetings has either three or four reference summaries. The remaining 69 meetings were used for training, which represent in total more than 103,000 training instances (or DA units), of which 8,271 are positives (8%). The multi-reference test set contains more than 28,000 instances.

The goal of a preliminary experiment was to devise a set of useful predictors from a full set of 1171. We performed feature selection by incrementally growing a log-linear model with order-0 features $f(\mathbf{x}, y_t)$ using a forward feature selection procedure similar to (Berger et al., 1996). Probably due to the imbalance between positive and negative examples, we found it more effective to rank candidate features by their gain in F-measure (through 5-fold cross validation on the entire training set). The increase in F by adding new features to the model is displayed in Table 4. This greedy search resulted in a set $\mathcal{S}$ of 217 features.

We now analyze the performance of different sequence models on our test set. The target length

of each summary was set to 12.7% of the number of words of the full document, which is the average on the entire training data (the average on the test data is 12.9%). In Table 5, we use an order-0 CRF to compare $\mathcal{S}$ against all features and various categorical groupings. Overall, we notice lexical predictors and statistics derived from them (e.g. LSA features) represent the most helpful feature group (.497), though all other features combined achieve a competitive performance (.476).

Table 6 displays performance for sequence models incorporating linear-chain features of increasing order k. Its second column indicates what criterion was used to rank utterances. In the case of 'pred', we used actual model $\{-1, 1\}$ predictions, which in all cases generated summaries much shorted than the allowable length, and produced poor performance. 'Costs' and 'norm-CRF' refer to the two ranking criteria presented in Section 5, and it is clear that the performance of CRF degrades with increasing orders without local normalization. While the contingency counts in Table 2 only hinted a limited benefit of linear-chain features, empirical results show the contrary—especially for order $k = 2$. However, the further increase of k cause overfitting, and skip-chain features seem a better way to capture non-local dependencies while keeping the number of model parameters relatively small. Overall, the addition of skip-chain edges to linear-chain models provide noticeable improvement in Pyramid scores. Our system that performed best on cross-validation data is a order-2 CRF with skip-chain transitions, which achieves a Pyramid score of $\mathcal{P} = .554$.

We now assess the significance of our results by comparing our best system against a lead summarizer, which always selects the first N utterances to match the predefined length, and human performance, which is obtained by leave-one-out comparisons among references (Table 7). Lastly, automatically generated "optimal" summaries using the procedure explained in (Nenkova and Passonneau, 2004b), by ranking document utterances by the number of model summaries in which they appear. It appears that our system is considerably better than the baseline, and achieves 91.3% of human performance in terms of Pyramid scores, and 83% if using ASR transcription. This last result is particularly positive if we consider our strong reliance on lexical features.

For completeness, we also included ROUGE (1, 2, and L) scores in Table 7, which were obtained using parameters defined for the DUC-05 evaluation. Since system summaries have on average approximately the same length as references, we only report recall measures of ROUGE (precision and F averages are within $\pm$ .002).[4] It may come as a surprise that our best system (both with ASR and true words) performs almost as well as humans; it seems more reasonable to conclude that, in our case, ROUGE has trouble discriminating between systems with moderately close performance. This seems to confirm our impression that content evaluation should be based on exact matches.

Finally, we performed a last experiment to compare our best system against Murray et al. (2005), who used the same test data, but constrained summary sizes in terms of number of DA units instead

FEATURE SET	$\mathcal{P}$
lexical	.471
IR	.415
lexical + IR	.497
acoustic	.407
structural/durational	.478
acoustic + structural/durational	.476
all features	.507
selected features ($\mathcal{S}$)	.515

Table 5: Pyramid score for each feature set.

MODEL	RANKING	$k = 1$	2	3
linear-chain BN	pred	.241	.267	.269
linear-chain BN	costs	.512	.519	.525
skip-chain BN	costs	.543	.549	.542
linear-chain CRF	pred	.326	.36	.348
linear-chain CRF	costs	.508	.475	.447
linear-chain CRF	norm-CRF	.53	.548	.54
skip-chain CRF	norm-CRF	.541	**.554**	.559

Table 6: Pyramid scores for different sequence models, where k stands for the order of linear-chain features. The value in bold is the performance of the model that was selected after a 5-fold cross validation on the training data, which obtained the highest $F_{\beta=1}$ score.

SUMMARIZER	$\mathcal{P}$	R-1	R-2	R-L
baseline	.188	.501	.210	.495
skip-chain CRF (transcript)	.554	.715	.442	.709
skip-chain CRF (ASR)	.504	.714	.42	.706
human	.607	.720	.477	.715
optimal	1	.791	.648	.788

Table 7: Pyramid, and average ROUGE scores for summaries produces by a baseline (lead summarizer), our best system, humans, and the optimal summarizer.

[4] Human performance with ROUGE was assessed by cross-validating reference summaries of each meeting (i.e., n references for a given meeting resulted in n evaluations against the other references). We used the same leave-one-out procedure with other summarizers, in order to get results comparable to humans.

of words (10% of the original document). Our system achieves .91 recall, .5 precision, and .64 F; the discrepancy between recall and precision is largely due to the much longer summary lengths. The best ROUGE-1 measure reported in (Murray et al., 2005) is .69 recall, which is significantly lower than ours according to confidence intervals.

9 Conclusion

An order-2 CRF with skip-chain dependencies derived from the automatic analysis of participant interaction was shown to outperform linear-chain BNs and CRFs, despite the incorporation in all cases of the same competitive set of predictors resulting from cross-validated feature selection. Compared to an order-0 CRF model, the absolute increase in performance is 3.9% (7.5% relative increase), which indicates that it is helpful to use skip-chain sequence models in the summarization task. Our best performing system reaches 91.3% of human performance, and scales relatively well on automatic speech recognition output.

Acknowledgments

This work has benefited greatly from suggestions and advice from Kathleen McKeown. I also would like to thank Jean Carletta, Steve Renals and Gabriel Murray for giving me access to their summarization corpus, Ani Nenkova for helpful discussions about summarization evaluation, Michael Collins, Daniel Ellis, Julia Hirschberg, and Owen Rambow for useful preliminary discussions, and three anonymous reviewers for their insightful comments on an earlier version of this paper.

References

A. Berger, S. Della Pietra, and V. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–72.

P. Boersma and D. Weenink. 2006. Praat: doing phonetics by computer. `http://www.praat.org/`.

J. Conroy, J. Schlesinger, J. Goldstein, and D. O'Leary. 2004. Left-brain/right-brain multi-document summarization. In *DUC 04 Conference Proceedings*.

H.P. Edmundson. 1968. New methods in automatic extracting. *Journal of the Association for Computing Machinery*, 16(2):264–285.

J. Finkel, T. Grenager, and C. Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proc. of ACL*, pages 363–370.

M. Galley, K. McKeown, J. Hirschberg, and E. Shriberg. 2004. Identifying agreement and disagreement in conversational speech: Use of bayesian networks to model pragmatic dependencies. In *Proc. of ACL*, pages 669–676.

M. Hearst. 1994. Multi-paragraph segmentation of expository text. In *Proc. of ACL*, pages 9–16.

A. Inoue, T. Mikami, and Y. Yamashita. 2004. Improvement of speech summarization using prosodic information. In *Proc. of Speech Prosody*.

A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters. 2003. The ICSI meeting corpus. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP–03)*.

J. Lafferty, A. McCallum, and F. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of ICML*, pages 282–289.

C.-Y. Lin. 2004. ROUGE: a package for automatic evaluation of summaries. In *Proc. of workshop on text summarization, ACL-04*.

I. Mani and M. Maybury. 1999. *Advances in Automatic Text Summarization*. MIT Press.

S. Maskey and J. Hirschberg. 2005. Comparing lexial, acoustic/prosodic, discourse and structural features for speech summarization. In *Proc. of Eurospeech*.

A. McCallum, D. Freitag, and F. Pereira. 2000. Maximum entropy markov models for information extraction and segmentation. In *Proc. of ICML*.

A. McCallum. 2002. MALLET: A machine learning for language toolkit. `http://mallet.cs.umass.edu`.

N. Mirghafori, A. Stolcke, C. Wooters, T. Pirinen, I. Bulyko, D. Gelbart, M. Graciarena, S. Otterson, B. Peskin, and M. Ostendorf. 2004. From switchboard to meetings: Development of the 2004 ICSI-SRI-UW meeting recognition system. In *Proc. of the International Conference of Spoken Language Processing (ICSLP–04)*.

G. Murray, S. Renals, J. Carletta, and J. Moore. 2005. Evaluating automatic summaries of meeting recordings. In *Proc. of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization*.

A. Nenkova and R. Passonneau. 2004a. Evaluating content selection in human- or machine-generated summaries: The pyramid scoring method. Technical Report CUCS-025-03, Columbia University, CS Department.

A. Nenkova and R. Passonneau. 2004b. Evaluating content selection in summarization: The pyramid method. In *Proc. of HLT/NAACL*, pages 145–152.

L. Rabiner. 1989. A tutorial on hidden markov models and selected applications in speech recogntion. *Proc. of the IEEE*, 77(2):257–286.

O. Rambow, L. Shrestha, J. Chen, and C. Lauridsen. 2004. Summarizing email threads. In *Proc. of HLT-NAACL*.

E. Schegloff and H. Sacks. 1973. Opening up closings. *Semiotica*, 7-4:289–327.

F. Sha and F. Pereira. 2003. Shallow parsing with conditional random fields. In *Proc. of NAACL*, pages 134–141.

L. Shrestha and K. McKeown. 2004. Detection of question-answer pairs in email conversations. In *Proc. of Coling*, pages 889–895.

E. Shriberg, R. Dhillon, S. Bhagat, J. Ang, and H. Carvey. 2004. The ICSI meeting recorder dialog act (MRDA) corpus. In *SIGdial Workshop on Discourse and Dialogue*, pages 97–100.

C. Sutton and A. McCallum. 2004. Collective segmentation and labeling of distant entities in information extraction. Technical Report TR # 04-49, University of Massachusetts, July.

K. Zechner. 2002. Automatic summarization of open domain multi-party dialogues in diverse genres. *Computational Liguistics*, 28(4):447–485.

L. Zhou and E. Hovy. 2005. Digesting virtual "geek" culture: The summarization of technical internet relay chats. In *Proc. of ACL*, pages 298–305.