Focused Meeting Summarization via Unsupervised Relation Extraction

Lu Wang
Department of Computer Science
Cornell University
Ithaca, NY 14853
luwang@cs.cornell.edu

Claire Cardie
Department of Computer Science
Cornell University
Ithaca, NY 14853
cardie@cs.cornell.edu

Abstract

We present a novel unsupervised framework for focused meeting summarization that views the problem as an instance of relation extraction. We adapt an existing in-domain relation learner (Chen et al., 2011) by exploiting a set of task-specific constraints and features. We evaluate the approach on a decision summarization task and show that it outperforms unsupervised utterance-level extractive summarization baselines as well as an existing generic relation-extraction-based summarization method. Moreover, our approach produces summaries competitive with those generated by supervised methods in terms of the standard ROUGE score.

1 Introduction

For better or worse, meetings play an integral role in most of our daily lives — they let us share information and collaborate with others to solve a problem, to generate ideas, and to weigh options. Not surprisingly then, there is growing interest in developing automatic methods for meeting summarization (e.g., Zechner (2002), Maskey and Hirschberg (2005), Galley (2006), Lin and Chen (2010), Murray et al. (2010a)). This paper tackles the task of focused meeting summarization, i.e., generating summaries of a particular aspect of a meeting rather than of the meeting as a whole (Carenini et al., 2011). For example, one might want a summary of just the DECISIONS made during the meeting, the ACTION ITEMS that emerged, the IDEAS discussed, or the HYPOTHESES put forth, etc.

Consider, for example, the task of summarizing the decisions in the dialogue snippet in Figure 1. The figure shows only the decision-related dialogue acts (DRDAs) — utterances associated with one or more decisions. Each DRDA is labeled numerically according to the decision it supports; so the first two utterances support DECISION 1 as do the final two utterances in the snippet. Manually constructed decision abstracts for each decision are shown at the bottom of the figure. These constitute the decision-focused summary for the snippet.

Notice that many portions of the DRDAs are not relevant to the decision itself: they often begin with phrases that identify the utterance within the discourse as potentially introducing a decision (e.g., “Maybe that could be”, “It seems like you’re gonna have”), but do not themselves describe the decision. We will refer to this portion of a DRDA (underlined in Figure 1) as the Decision Cue.

Moreover, the decision cue is generally directly followed by the actual Decision Content (e.g., “be a little apple”, “have rubber cases”). Decision Content phrases are denoted in Figure 1 via italics and square brackets. Importantly, it is just the decision content portion of the utterance that should be considered for incorporation into the focused summary.

1These are similar, but not completely equivalent, to the decision dialogue acts (DDAs) of (Bui et al., 2009), (Fernández et al., 2008), (Frampton et al., 2009).

2Murray et al. (2010b) show that users much prefer abstractive summaries over extracts when the text to be summarized is a conversation. In particular, extractive summaries drawn from group conversations can be confusing to the reader without additional context; and the noisy, error-prone, disfluent text of speech transcripts is likely to result in extractive summaries with low readability.
This paper presents an unsupervised framework for focused meeting summarization that supports the generation of abstractive summaries. (Note that we do not currently generate actual abstracts, but rather aim to identify those Content phrases that should comprise the abstract.) In contrast to existing approaches to focused meeting summarization (e.g., Purver et al. (2007), Fernández et al. (2008), Bui et al. (2009)), we view the problem as an information extraction task and hypothesize that existing methods for domain-specific relation extraction can be modified to identify salient phrases for use in generating abstractive summaries.

Very generally, information extraction methods identify a lexical “trigger” or “indicator” that evokes a relation of interest and then employ syntactic information, often in conjunction with semantic constraints, to find the “target phrase” or “argument constituent” to be extracted. Relation instances, then, are represented by indicator-argument pairs (Chen et al., 2011).

Figure 1 shows some possible indicator-argument pairs for identifying the Decision Content phrases in the dialogue sample. Content indicator words are shown in italics; the Decision Content target phrases are the arguments. For example, in the fourth DRDA, “require” is the indicator, and “rubber buttons” and “rubber case” are both arguments. Although not shown in Figure 1, it is also possible to identify relations that correspond to the Decision Cue phrases.³

Specifically, we focus on the task of decision summarization and, as in previous work in meeting summarization (e.g., Fernández et al. (2008), Wang and Cardie (2011)), assume that all decision-related utterances (DRDAs) have been identified. We adapt the unsupervised relation learning approach of Chen et al. (2011) to separately identify relations associated with decision cues vs. the decision content within DRDAs by defining a new set of task-specific constraints and features to take the place of the domain-specific constraints and features of the original model. Output of the system is a set of extracted indicator-argument decision content relations (see the “OUR METHOD” sample summary of Table 6) that can be used as the basis of the decision abstract.

We evaluate the approach (using the AMI corpus (Carletta et al., 2005)) under two input settings — in the True Clusterings setting, we assume that the DRDAs for each meeting have been perfectly grouped according to the decision(s) each supports; in the System Clusterings setting, an automated system performs the DRDA-decision pairing. The results show that the relation-based summarization approach outperforms two extractive summarization baselines that select the longest and the most representative utterance for each decision, respectively. (ROUGE-1 F score of 37.47% vs. 32.61% and 33.32% for the baselines given the True Clusterings of DRDAs.) Moreover, our approach performs admirably in comparison to two supervised learning alternatives (scores of 35.61% and 40.87%) that aim to identify the important tokens to include in the decision abstract given the DRDA clusterings. In contrast to our approach which is transferable to different domains or tasks, these methods would require labeled data for retraining for each new meeting corpus.

³Consider, for example, the phrases underlined in the sixth and seventh DRDAs. “I mean” and “shall we” are two typical Decision Cue phrases where “mean” and “shall” are possible indicators with “I” and “we” as their arguments, respectively.
Finally, in order to compare our approach to another relation-based summarization technique, we modify the multi-document summarization system of Hachey (2009) to the single-document meeting scenario. Here again, our proposed approach performs better (37.47% vs. 34.69%). Experiments under the System Clusterings setting produce the same overall results, albeit with lower scores for all of the systems and baselines.

In the remainder of the paper, we review related work in Section 2 and give a high-level description of the relation-based approach to focused summarization in Section 3. Sections 4, 5, and 6 present the modifications to the Chen et al. (2011) relation extraction model required for its instantiation for the meeting summarization task. Sections 7 and 8 provide our experimental setup and results.

2 Related Work

Most research on spoken dialogue summarization attempts to generate summaries for full dialogues (Carenini et al., 2011). Only recently, however, has the task of focused summarization, and decision summarization, in particular, been addressed. Fernández et al. (2008) and Bui et al. (2009) employ supervised learning methods to rank phrases or words for inclusion in the decision summary. In comparison, Fernández et al. (2008) find that the phrase-based approach yields better recall than token-based methods, concluding that phrases have the potential to support better summaries. Input to their system, however, is narrowed down (manually) from the full set of DRDAs to the subset that is useful for summarization. In addition, they evaluate their system w.r.t. informative phrases or words that have been manually annotated within this DRDA subset. We are instead interested in comparing our extracted relations to the abstractive summaries.

In contrast to our phrase-based approach, we previously explored a collection of supervised and unsupervised learning methods for utterance-level (i.e., dialogue act) and token-level decision summarization (Wang and Cardie, 2011). We adopt here the two unsupervised baselines (utterance-level summaries) from that work for use in our evaluation. We further employ their supervised summarization methods as comparison points for token-level summarization, adding additional features for consistency with the other approaches in the evaluation. Murray et al. (2010a) develop an integer linear programming approach for focused summarization at the utterance-level, selecting sentences that cover more of the entities mentioned in the meeting as determined through the use of an external ontology.

The most relevant previous work is Hachey (2009), which uses relational representations to facilitate sentence-ranking for multi-document summarization. The method utilizes generic relation extraction to represent the concepts in the documents as relation instances; summaries are generated based on a set cover algorithm that selects a subset of the sentences that best cover the weighted concepts. Thus, the goal of Hachey’s approach is sentence extraction rather than phrase extraction. Although his relation extraction method, like ours (see Section 4), is probabilistic and unsupervised (he uses Latent Dirichlet Allocation (Blei et al., 2003)), the relations are limited to pairs of named-entities, which is not appropriate for our decision summarization setting. Nevertheless, we will adapt his approach for comparison with our relation-based summarization technique and include it for evaluation.

3 Focused Summarization as Relation Extraction

Given the DRDAs for each meeting grouped (not necessarily correctly) according to the decisions they support, we put each cluster of DRDAs (ordered according to time within the cluster) into one “decision document”. The goal will be to produce one decision abstract for each such decision document. We obtain constituent and dependency parses using the Stanford parser (Klein and Manning, 2003; de Marneffe et al., 2006). With the corpus of constituent-parsed decision documents as the input, we will use and modify Chen et al. (2011)’s system to identify decision cue relations and decision content relations for each cluster. (Section 6 will make clear how the learned decision cue relations will be used to identify decision content relations.) The salient decision content relation instances will be returned as decision summary com-

4Other unsupervised relation learning methods might also be appropriate (e.g., Open IE (Banko et al., 2007)), but they generally model relations between pairs of entities and group relations only according to lexical similarity.
ponents.

Designed for in-domain relation discovery from standard written texts (e.g., newswire), however, the Chen et al. (2011) system cannot be applied to our task directly. In our setting, for example, neither the number of relations nor the relation types is known in advance.

In the following sections, we describe the modifications needed for the spoken meeting genre and decision-focused summarization task. In particular, Chen et al. (2011) provide two mechanisms that allow for this type of tailoring: the feature set used to cluster potential relation instances into groups/types, and a set of global constraints that characterize the general qualities (e.g., syntactic form, prevalence, discourse behavior) of a good relation for the task.

4 Model

In this section, we describe the Chen et al. (2011) probabilistic relation learning model used for both Decision Cue and Decision Content relation extraction. The parameter estimation and constraint encoding through posterior inference are presented in Section 5.

The relation learning model takes as input clusters of DRDAs, sorted according to utterance time and concatenated into one decision document. We assume one decision will be made per document. The goal for the model is to explain how the decision documents are generated from the latent relation variables. The posterior regularization technique (Section 5) biases inference to adhere to the declarative constraints on relation instances. In general, instead of extracting relation instances strictly satisfying a set of human-written rules, features and constraints are designed to allow the model to reveal diverse relation types and to ensure that the identified relation instances are coherent and meaningful. For each decision document, we select the relation instance with highest probability for each relation type and concatenate them to form the decision summary.

We restrict the eligible indicators to be a noun or verb, and eligible arguments to be a noun phrase (NP), prepositional phrase (PP) or clause introduced by “to” (S). Given a pre-specified number of relation types $K$, the model employs a set of features $\phi^f(w)$ and $\phi^a(x)$ (see Section 6) to describe the indicator word $w$ and argument constituent $x$. Each relation type $k$ is associated with a set of feature distributions $\theta_k$, and a location distribution $\lambda_k$. $\theta_k$ include four parameter vectors: $\theta_k^f$ for indicator words, $\theta_k^a$ for non-indicator words, $\theta_k^b$ for argument constituents, and $\theta_k^{ba}$ for non-argument constituents. Each decision document is divided into $L$ equal-length segments and the location parameter vector $\lambda_k$ describes the probability of relation $k$ arising from each segment. The plate diagram for the model is shown in Figure 2. The generative process and likelihood of the model are shown in Appendix A.

5 Parameter Estimation and Inference via Posterior Regularization

In order to specify global preferences for the relation instances (e.g. the syntactic structure of the expressions), we impose inequality constraints on expectations of the posterior distributions during inference (Graca et al., 2008).

5.1 Variational inference with Constraints

Suppose we are interested in estimating the posterior distribution $p(\theta, z|x)$ of a model in general, where $\theta$, $z$ and $x$ are parameters to estimate, latent variables and observations, respectively. We aim to find a distribution $q(\theta, z) \in Q$ that minimizes the KL-divergence to the true posterior

$$KL(q(\theta, z)||p(\theta, z|x))$$

(1)
A mean-field assumption is made for variational inference, where \( q(\theta, z) = q(\theta)q(z) \). Then we can minimize Equation 1 by performing coordinate descent on \( q(\theta) \) and \( q(z) \). Now we intend to have fine-level control on the posteriors to induce meaningful semantic parts. For instance, we would like most of the extracted relation instances to satisfy a set of pre-defined syntactic patterns. As presented in (Graca et al., 2008), a general way to put constraints on posterior \( q \) is through bounding expectations of given functions: \( E_q[f(z)] \leq b \), where \( f(z) \) is a deterministic function of \( z \), and \( b \) is a pre-specified threshold. For instance, define \( f(z) \) as a function to count the number of generated relation instances that meet the pre-defined syntactic patterns, then most of the extracted relation instances will have the desired syntactic structures.

By using the mean-field assumption, the model in Section 4 is factorized as

\[
q(\theta, \lambda, z, i, a) = \\
\prod_{k=1}^{K} q(\lambda_k; \tilde{\lambda}_k)q(\theta_k; \tilde{\theta}_k)q(\theta_{kb}; \tilde{\theta}_{kb})q(\theta_{ka}; \tilde{\theta}_{ka}) \\
\times \prod_{d=1}^{D} q(z_{d,k}, i_{d,k}, a_{d,k}; \tilde{c}_{d,k})
\]

The constraints are encoded in the inequalities \( E_q[f(z, i, a)] \geq b \) or \( E_q[f(z, i, a)] \leq b \), and affect the inference as described above. Updates for the parameters are discussed in Appendix B.

5.2 Task-Specific Constraints.

We define four types of constraints for the decision relation extraction model.

Syntactic Constraints. Syntactic constraints are widely used for information extraction (IE) systems (Snow et al., 2005; Banko and Etzioni, 2008), as it has been shown that most relations are expressed via a small number of common syntactic patterns. For each relation type, we require at least 80%\(^{5}\) of the induced relation instances in expectation to match one of the following syntactic patterns:

- The indicator is a verb and the argument is a prepositional phrase or a clause starting with “to”. The indicator and the argument have the same parent in the constituent parsing tree.
- The indicator is a noun and is the headword of a noun phrase, and the argument is a prepositional phrase. The noun phrase with the indicator as its headword and the argument have the same parent in the constituent parsing tree.

For relation \( k \), let \( f(z_k, i_k, a_k) \) count the number of induced indicator \( i_k \) and argument \( a_k \) pairs that match one of the patterns above, and \( b \) is set to \( 0.8D \), where \( D \) is the number of decision documents. Then the syntactic constraint is encoded in the inequality

\[
E_q[f(z_k, i_k, a_k)] \geq b.
\]

Prevalence Constraints. The prevalence constraint is enforced on the number of times a relation is instantiated, in order to guarantee that every relation has enough instantiations across the corpus and is task-relevant. Again, we require each relation to have induced instances in at least 80% of decision documents.

Occurrence Constraints. Diversity of relation types is enforced through occurrence constraints. In particular, for each decision document, we restrict each word to trigger at most two relation types as indicator and occur at most twice as part of a relation’s argument in expectation. An entire span of argument constituent can appear in at most one relation type.

Discourse Constraints. The discourse constraint captures the insight that the final decision on an issue is generally made, or at least restated, at the end of the decision-related discussion. As each decision document is divided into four equal parts, we restrict 50% of the relation instances to be from the last quarter of the decision documents.

6 Features

Table 1 lists the features we use for discovering both the decision cue relations and decision content relations. We start with a collection of domain-independent Basic Features shown to be useful in relation extraction (Banko and Etzioni, 2008; Chen et al., 2011). Then we add Meeting Features, Structural Features and Semantic Features that have been found to be good predictors for decision detection (Hsueh and Moore, 2007) or meeting and decision summarization (Gal-
Table 1: Features for Decision Cue and Decision Content relation extraction. All features, except the last type of features, are used for both the indicator and argument. (An Adjacency Pair (AP) is an important conversational analysis concept (Schegloff and Sacks, 1973). In the AMI corpus, an AP pair consists of a source utterance and a target utterance, produced by different speakers.)

<table>
<thead>
<tr>
<th>Basic Features</th>
<th>Decision Cue Relations</th>
<th>Relation Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>unigram (stemmed)</td>
<td>Group Wrap-up / Recap</td>
<td>we have, we are, we</td>
</tr>
<tr>
<td>part-of-speech (POS)</td>
<td>Personal Explanation</td>
<td>say, I mean, I think</td>
</tr>
<tr>
<td>constituent label</td>
<td>Suggestion</td>
<td>I, I guess, I (would)</td>
</tr>
<tr>
<td>(NP, VP, S/SBAR (start</td>
<td>Final Decision</td>
<td>say) do we, we (could/should)</td>
</tr>
<tr>
<td>with “to”)</td>
<td></td>
<td>do</td>
</tr>
<tr>
<td>dependency label</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Sample Decision Cue relation instances. The words in parentheses are filled for illustration purposes, while they are not part of the relation instances.

<table>
<thead>
<tr>
<th>Discourse Features</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>clause position (first,</td>
<td>position to the first</td>
</tr>
<tr>
<td>second, other)</td>
<td>decision cue relation if</td>
</tr>
<tr>
<td></td>
<td>any (before, after)</td>
</tr>
</tbody>
</table>

Table 3: Additional features for Decision Content relation extraction, inspired by Decision Cue relations. Both indicator and argument use those features.

of annotations. The 129 scenario-driven meetings involve four participants playing different roles on a design team. Importantly, the corpus includes a short (usually one-sentence), manually constructed abstract summarizing each decision discussed in the meeting. In addition, all of the dialogue acts that support (i.e., are relevant to) each decision are annotated as such. We use the manually constructed decision abstracts as gold-standard summaries.

System Inputs. We consider two system input settings. In the True Clusterings setting, we use the AMI annotations to create perfect partitionings of the DRDAs for input to the summarization system; in the System Clusterings setting, we employ a hierarchical agglomerative clustering algorithm used for this task in previous work (Wang and Cardie, 2011). The Wang and Cardie (2011) clustering method groups DRDAs according to their LDA topic distribution similarity. As better approaches for DRDA clustering become available, they could be employed instead.

Evaluation Metrics. We use the widely accepted ROUGE (Lin and Hovy, 2003) evaluation measure. We adopt the ROUGE-1 and ROUGE-SU4 metrics from (Hachey, 2009), and also use ROUGE-2. We choose the stemming option of the ROUGE software at http://berouge.com/ and remove stopwords from both the system and gold-standard summaries.

Training and Parameters. The Dirichlet hyperparameters are set to 0.1 for the priors. When training the model, ten random restarts are performed and each run stops when reaching a convergence threshold ($10^{-5}$). Then we select the posterior with...
the lowest final free energy. For the parameters used in posterior constraints, we either adopt them from (Chen et al., 2011) or choose them arbitrarily without tuning in the spirit of making the approach domain-independent.

We compare our decision summarization approach with (1) two unsupervised baselines, (2) the unsupervised relation-based approach of Hachey (2009), (3) two supervised methods, and (4) an upperbound derived from the gold standard decision abstracts.

**The Longest DA Baseline.** As in Riedhammer et al. (2010) and Wang and Cardie (2011), this baseline simply selects the longest DRDA in each cluster as the summary. Thus, this baseline performs utterance-level decision summarization. Although it’s possible that decision content is spread over multiple DRDAs in the cluster, this baseline and the next allow us to determine summary quality when summaries are restricted to a single utterance.

**The Prototype DA Baseline.** Following Wang and Cardie (2011), the second baseline selects the decision cluster prototype (i.e., the DRDA with the largest TF-IDF similarity with the cluster centroid) as the summary.

**The Generic Relation Extraction (GRE) Method of Hachey (2009).** Hachey (2009) presents a generic relation extraction (GRE) for multi-document summarization. Informative sentences are extracted to form summaries instead of relation instances. Relation types are discovered by Latent Dirichlet Allocation, such that a probability is output for each relation instance given a topic (equivalent to relation). Their relation instances are named entity(NE)-mention pairs conforming to a set of pre-specified rules. For comparison, we use these same rules to select noun-mention pairs rather than NE-mention pairs, which is better suited to meetings, which do not contain many NEs.6

**Supervised Learning (SVMs and CRFs).** We also compare our approach to two supervised learning methods — Support Vector Machines (Joachims, 1998) with RBF kernel and order-1 Conditional Random Fields (Lafferty et al., 2001) — trained using the same features as our system (see Tables 1 and 3) to identify the important tokens to include in the decision abstract. Three-fold cross validation is conducted for both methods.

**Upperbound.** We also compute an upperbound that reflects the gap between the best possible extractive summaries and the human-written abstracts according to the ROUGE score: for each cluster of DRDAs, we select the words that also appear in the associated decision abstract.

### 8 Results and Discussion

Table 4 illustrates that, using True (DRDA) Clusterings our method outperforms the two baselines and the generic relation extraction (GRE) based system in terms of F score in ROUGE-1 and ROUGE-SU4 with varied numbers of relations. Note that for GRE based approach, we only list out their best results for utterance-level summarization. If using the salient relation instances identified by GRE as the summaries, the ROUGE results will be significantly better than the second, so we only report the best results for their system in this paper.
lower. When measured by ROUGE-2, our method still have better or comparable performances than other unsupervised methods. Moreover, our system achieves F scores in between those of the supervised learning methods, performing better than the CRF in both recall and F score. The recall score for the upperbound in ROUGE-1, on the other hand, indicates that there is still a wide gap between the extractive summaries and human-written abstracts: without additional lexical information (e.g., semantic class information, ontologies) or a real language generation component, recall appears to be a bottleneck for extractive summarization methods that select content only from decision-related dialogue acts (DRDAs).

Results using the System Clusterings (Table 5) are comparable, although all of the system and baseline scores are much lower. Supervised methods get the best F scores largely due to their high precision; but our method attains the best recall in ROUGE-1.

<table>
<thead>
<tr>
<th>System Clusterings</th>
<th>R-1</th>
<th>R-2</th>
<th>R-SU4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Baselines</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Longest DA</td>
<td>17.06</td>
<td>11.64</td>
<td>13.84</td>
</tr>
<tr>
<td>Prototype DA</td>
<td>18.14</td>
<td>10.11</td>
<td>12.98</td>
</tr>
<tr>
<td><strong>GRE</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 topics</td>
<td>17.10</td>
<td>9.76</td>
<td>12.40</td>
</tr>
<tr>
<td>10 topics</td>
<td>16.28</td>
<td>10.03</td>
<td>12.35</td>
</tr>
<tr>
<td>15 topics</td>
<td>16.54</td>
<td>10.90</td>
<td>13.04</td>
</tr>
<tr>
<td>20 topics</td>
<td>17.25</td>
<td>8.99</td>
<td>11.80</td>
</tr>
<tr>
<td><strong>Supervised Methods</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRF</td>
<td>47.36</td>
<td>15.34</td>
<td>23.18</td>
</tr>
<tr>
<td>SVM</td>
<td>39.50</td>
<td>18.49</td>
<td>25.19</td>
</tr>
<tr>
<td><strong>Our Method</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 Relations</td>
<td>16.12</td>
<td>18.93</td>
<td>17.41</td>
</tr>
<tr>
<td>10 Relations</td>
<td>16.27</td>
<td>18.93</td>
<td>17.50</td>
</tr>
<tr>
<td>15 Relations</td>
<td>16.42</td>
<td>19.14</td>
<td>17.68</td>
</tr>
<tr>
<td>20 Relations</td>
<td>16.75</td>
<td>18.25</td>
<td>17.47</td>
</tr>
</tbody>
</table>

Table 5: ROUGE-1 (R-1), ROUGE-2 (R-2) and ROUGE-SU4 (R-SU4) scores for summaries produced by the baselines, GRE (Hachey, 2009)’s best results, the supervised methods and our method — all with system clusterings.

**Discussion.** To better exemplify the summaries generated by different systems, sample output for each method is shown in Table 6. The GRE system uses an approximate algorithm for set cover extraction, we list the first three selected DRDA in order. We see from the table that utterance-level extractive summaries (Longest DA, Prototype DA, GRE) make more coherent but still far from concise and compact abstracts. On the other hand, the supervised methods (SVM, CRF) that produce token-level extracts better identify the overall content of the decision abstract. Unfortunately, they require human annotation in the training phase; in addition, the output is ungrammatical and lacks coherence. In comparison, our system presents the decision summary in the form of phrase-based relations that provide a relatively comprehensive expression.

**9 Conclusions**

We present a novel framework for focused meeting summarization based on unsupervised relation extraction. Our approach is shown to outperform unsupervised utterance-level extractive summarization baselines as well as an existing generic relation-extraction-based summarization method. Our approach also produces summaries competitive with those generated by supervised methods in terms of the standard ROUGE score. Overall, we find that relation-based methods for focused summarization have potential as a technique for supporting the generation of abstractive decision summaries.

**Acknowledgments** This work was supported in part by National Science Foundation Grants IIS-0968450 and IIS-1111176, and by a gift from Google.
References


Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure trees. In LREC.


Gabriel Murray and Giuseppe Carenini. 2008. Summarizing spoken and written conversations. In Proceed-


Appendix A Generative Process

The entire generative process is as follows (“Dir” and “Mult” refer to the Dirichlet distribution and multinomial distribution):

1. For each relation type k:
   (a) For each indicator feature $\phi^i$, draw feature distributions $\theta_{k,i,\phi^i}, \theta_{k,\phi^i} \sim \text{Dir}(\theta_0)$
   (b) For each argument feature $\phi^a$, draw feature distributions $\theta_{k,i,\phi^a}, \theta_{k,\phi^a} \sim \text{Dir}(\theta_0)$
   (c) Draw location distribution $\lambda_k \sim \text{Dir}(\lambda_0)$
2. For each relation type k and decision document d:
   (a) Select decision document segment $s_{d,k} \sim \text{Mult}(\lambda_k)$

(b) Select DRDA $z_{d,k}$ uniformly from segment $s_{d,k}$, and indicator $i_{d,k}$ and argument constituent $a_{d,k}$ uniformly from DRDA $z_{d,k}$

3. For each indicator word $w$ in every decision document $d$:
   (a) For each indicator feature $\phi^i(w)$ \sim Mult($\frac{1}{2}\prod_{k=1}^{K} \theta_{k,\phi^i}$), where $\theta_{k,i,\phi^i}$ is $\theta_{k,\phi^i}$ if $i_{d,k} = w$ and $\theta_{k,\phi^i}$ otherwise. $Z$ is the normalization factor.

4. For each argument constituent $x$ in every decision document $d$:
   (a) For each indicator feature $\phi^a(x)$ \sim Mult($\frac{1}{2}\prod_{k=1}^{K} \theta_{k,\phi^a}$), where $\theta_{k,a,\phi^a}$ is $\theta_{k,\phi^a}$ if $a_{d,k} = x$ and $\theta_{k,a,\phi^a}$ otherwise. $Z$ is the normalization factor.

Given $\theta_0$ and $\lambda_0$, The joint distribution of a set of feature parameters $\theta$, the location distributions $\lambda$, a set of DRDAs $z$, and the selected indicators $i$ and arguments $a$ is:

$$P(\theta, \lambda, z, i, a; \theta_0, \lambda_0) = \prod_{k=1}^{K} P(\theta_k^i; \theta_0) P(\theta_k^a; \theta_0) P(\theta_k^z; \theta_0) P(\lambda_k; \lambda_0)$$

$$\times \left( \prod_{d=1}^{D} P(i_{d,k}; z_{d,k}) P(a_{d,k}; z_{d,k}) P(s_{d,k}; \lambda_k) \right)$$

$$\times \left( P(w = i_{d,k}; \theta_k^i) \prod_{w \neq i_{d,k}} P(w; \theta_k^i) \right)$$

$$\times \left( P(x = a_{d,k}; \theta_k^a) \prod_{x \neq a_{d,k}} P(x; \theta_k^a) \right)$$

Appendix B Updates for the Parameters

The constraints put on the posterior will only affect the update for $q(z)$. For $q(\theta)$, the update is

$$q(\theta) = \arg\min_{q(\theta)} \text{KL}(q(\theta)||q'(\theta)),$$

where $q'(\theta) \propto \exp E_{q(z)}[\log p(\theta, z, x)]$, and $q(\theta)$ is updated to $q'(\theta)$. For $q(z)$, the update is

$$q(z) = \arg\min_{q(z)} \text{KL}(q(z)||q'(z))$$

s.t. $E_{q(z)}[f(z)] \leq b_c, \forall c \in C$

where $q'(z) \propto \exp E_{q(\theta)}[\log p(\theta, z, x)]$. Equation 4 is easily solved via the dual (Graca et al., 2008) (Chen et al., 2011).