Abstractive Meeting Summarization Using Dependency Graph Fusion

ABSTRACT

Automatic summarization techniques on meeting conversations developed so far have been primarily extractive, resulting in poor summaries. To improve this, we propose an approach to generate abstractive summaries by fusing important content from several utterances. Any meeting is generally comprised of several discussion topic segments. For each topic segment within a meeting conversation, we aim to generate a one sentence summary from the most important utterances using an integer linear programming-based sentence fusion approach. Experimental results show that our method can generate more informative summaries than the baselines.

Categories and Subject Descriptors
I.2.7 [Artificial Intelligence]: Natural Language Processing—Language generation

Keywords
Abstractive meeting summarization; Integer linear programming

1. INTRODUCTION

Meeting summarization helps both participants and non-participants by providing a short and concise snapshot of the most important content discussed in the meetings. A recent study revealed that people generally prefer abstractive summaries [4]. Table 1 shows the human-written abstractive summaries along with the human-generated extractive summaries from a meeting transcription. As can be seen, the utterances are highly noisy and contain unnecessary information. Even if an extractive summarizer can accurately classify these utterances as “important” and present them to a reader, it is hard to read and synthesize information from such utterances. In contrast, human written summaries are compact and readable.

We propose an automatic way of generating short and concise abstractive summaries of meetings. Any meeting conversation includes dialogues on several topics. For example, in Table 1, the participants converse on two topics: design features and selling prices. Given the most important sentences within a topic segment, our goal is to generate a one-sentence summary from each segment and appending them to form a comprehensive summary of the meeting. Moreover, we also aim to generate summaries that resemble human-written summaries in terms of writing style.

To aggregate the information from multiple utterances, we adapt an existing integer linear programming (ILP) based fusion technique [1]. The fusion technique is based on the idea of merging dependency parse trees of the utterances. The trees are merged on the common nodes that are represented by the word and parts-of-speech (POS) combination. Each edge of the merged structure is represented as a variable in the ILP objective function, and the solution will decide whether the edge has to be preserved or discarded. We modify the technique by introducing an anaphora resolution step and also an ambiguity resolver that takes the context of words into account. Further, to solve the ILP, we introduce several constraints, such as desired length of the output, etc.

To the best of our knowledge, our work is the first to address the problems of readability, grammaticality and content selection jointly for meeting summary generation without employing a template-based approach. We conduct experiments on the AMI corpus that consists of meeting transcripts and show that our best method out-performs extractive model significantly on ROUGE-2 scores (0.048 vs 0.026).

2. PROPOSED APPROACH

Dependency fusion on meeting data requires an algorithm that is robust for noisy data as utterances often have disfluencies. Our work applies fusion to all the important utterances within the topic segment to generate the best sub-tree that satisfies the constraints and maximizes the objective function of the optimization problem. Anaphora resolution step replaces pronouns with the original nouns in the previous utterance that they refer to in order to increase the chances of merging. Consider the following utterances:

"Um, as you can see it’s supposed to be original "

Without pronoun resolution, these two utterances cannot be merged. Once we apply anaphora resolution, it in the second utterance is modified to a new remote control and then both the utterances are fused into a common structure. The utterances are parsed using the Stanford dependency parser. Every individual utterance has an explicit ROOT node. We add two dummy nodes in the graph – the

Table 1: Two sets of extractive summaries and gold standard human generated abstractive summaries from a meeting (Set 2 follows Set 1).

<table>
<thead>
<tr>
<th>Set</th>
<th>Human-generated extractive summary</th>
<th>Abstractive summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>B: Like how much does, you know, a remote control cost.</td>
<td>D: kay trendy probably means something other than just basic</td>
</tr>
<tr>
<td>2</td>
<td>B: So I don’t know how good a remote control that would get you. Um.</td>
<td>D: I’m thinking the price might appeal to a certain market in one region, whereas in another it’ll be different, so</td>
</tr>
<tr>
<td>3</td>
<td>D: Well twenty five Euro, I mean that’s um that’s about like eighteen pounds or something.</td>
<td>A: well fifteen basically might be a reasonable price</td>
</tr>
<tr>
<td>4</td>
<td>B: Like how much does, you know, a remote control cost.</td>
<td>D: um as well as uh keypads and s symbols</td>
</tr>
<tr>
<td>5</td>
<td>D: Well right away I’m wondering if there’s um th th uh, like with D_V_D players, if there are zones.</td>
<td>A: Cause you have more complicated characters like European languages, then you need more buttons</td>
</tr>
<tr>
<td>6</td>
<td>A: Cause you have more complicated characters like European languages, then you need more buttons.</td>
<td>A: Cause you have more complicated characters like European languages, then you need more buttons</td>
</tr>
<tr>
<td>7</td>
<td>D: Well right away I’m wondering if there’s um th th uh, like with D_V_D players, if there are zones.</td>
<td>A: Cause you have more complicated characters like European languages, then you need more buttons</td>
</tr>
</tbody>
</table>

The project manager talked about the project finances and selling prices.

B: So I don’t know how good a remote control that would get you. Um.

D: kay trendy probably means something other than just basic

D: I’m thinking the price might appeal to a certain market in one region, whereas in another it’ll be different, so

D: Well right away I’m wondering if there’s um th th uh, like with D_V_D players, if there are zones.

B: Like how much does, you know, a remote control cost.

D: Well twenty five Euro, I mean that’s um that’s about like eighteen pounds or something.

D: um as well as uh keypads and s symbols.

B: So I don’t know how good a remote control that would get you. Um.

D: This is this gonna to be like the premium product kinda thing or

A: well fifteen basically might be a reasonable price.

A: Cause you have more complicated characters like European languages, then you need more buttons.

A: Cause you have more complicated characters like European languages, then you need more buttons.

A: Cause you have more complicated characters like European languages, then you need more buttons.

A: Cause you have more complicated characters like European languages, then you need more buttons.

A: Cause you have more complicated characters like European languages, then you need more buttons.

A: Cause you have more complicated characters like European languages, then you need more buttons.

A: Cause you have more complicated characters like European languages, then you need more buttons.

A: Cause you have more complicated characters like European languages, then you need more buttons.

A: Cause you have more complicated characters like European languages, then you need more buttons.
In order to take this fact into account, we introduce the term end of any segment, generally, more important discussions might and only the relations that are more dominant from a node will be. This term assigns the importance of grammatical relations to a node and the label of an edge, respectively. We maximize the following objective function:

\[
\sum_{x} p_{d,g,l} \cdot P(l \mid g) \cdot I(d) \cdot \frac{p_{\text{end}}}{N}
\]  

As shown in Equation (1), we introduce three different terms: \(p(l \mid g)\), \(I(d)\) and \(\frac{p_{\text{end}}}{N}\). Each relation in a dependency graph consists of the governing node, the dependent node and the relation type. The term \(p(l \mid g)\) denotes the probabilities of the labels given a governor node, \(g\). For every node (word and POS) in the entire corpus, the probabilities are represented as the ratio of the sum of the frequency of a particular label and the sum of the frequencies of all the labels emerging from a node. In this work, we calculate these values using Reuters corpora [5] to obtain dominant relations from non-conversational style of text. For example, Table 2 shows the probabilities of outgoing edges from a node, ("produced/VBN").

![Figure 1: A merged dependency graph structure – edges in blue bold arrows to be retained to generate the summary for each topic segment.](image)

Table 2: Probabilities of relations from "produced/VBN."

<table>
<thead>
<tr>
<th>auxpass</th>
<th>nsubjpass</th>
<th>aux</th>
<th>prep_with</th>
<th>agent</th>
<th>prep_in</th>
<th>advmod</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.286</td>
<td>0.241</td>
<td>0.241</td>
<td>0.031</td>
<td>0.326</td>
<td>0.037</td>
<td>0.034</td>
</tr>
</tbody>
</table>

where \(N\) and \(p_x\) denote the total number of extracted utterances in a segment and the position of the utterance (the edge \(x\) belongs to) in the set of \(N\) utterances, respectively.

In order to solve the above ILP problem, we impose a number of constraints. Some of the constraints have been directly adapted from the original ILP formulation [1]. For example, we use the same constraints for restricting one incoming edge per node, as well as we impose the connectivity constraint to ensure a connected graph structure. Further, we restrict the subtree to have just one start and one end edge. This helps in preserving one ROOT node, as well as it limits to one end node for the generated subtree. We also limit the generated subtree to have a maximum of 15 nodes that controls the length of the summary sentence. We also add few linguistic constraints that ensure the coherence of the output such as every node can have maximum of one determinant, etc. We also impose constraints to prevent cycles in the graph structure, otherwise finding the best path from start and end nodes might be difficult. The final graph is linearized to obtain a coherent sentence. In the linearization process, we order the nodes based on their original ordering in the utterance.

3. EXPERIMENTAL RESULTS

The AMI Meeting corpus contains 20 meeting transcripts in the test set along with their corresponding abstractive (human-written) summaries as well as the annotations of topic segments. ROUGE is used to compare content selection of several approaches. We compared the content selection of our approach to an extractive summarizer [3], which works as a baseline. We also compared our model without using anaphora resolution to see the impact of resolving pronouns. All the summaries were compared against the human-written summaries as reference. The results in Table 3 show that our method outperforms the other techniques on both ROUGE-2 (R2) and ROUGE-SU4 (R-SU4) recall scores. Moreover, we computed a coarse estimate of grammaticality using the log-likelihood score (LL) from the parser. Our technique significantly outperforms the extractive method. In future work, we plan to design an end-to-end framework for summary generation from meetings.

4. REFERENCES