Formalizing CST-Based Content Selection Operations*

Maria Lucía Castro Jorge and Thiago Alexandre Salgueiro Pardo

Núcleo Interinstitucional de Lingüística Computacional (NILC)
Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo
{mluciacj,taspardo}@icmc.usp.br

Abstract. This paper presents the definition and formalization of content selection operations based on CST (Cross-document Structure Theory) for multidocument summarization purposes.

Keywords: Multidocument summarization, CST, user preferences.

1 Introduction

Multidocument summarization (MDS) is a research area that intends to automatically produce a summary from a group of texts on the same topic [3]. A good MDS system must be able to produce a single summary telling the main points of the story or some parts of it for an interested reader, as background information or the evolution in time of some event. Therefore, content selection is an important task for MDS.

In this paper, we define and formalize content selection operations based on Cross-document Structure Theory (CST) [5], a multidocument theory/model that intends to represent the relatedness of a group of texts on the same topic. Our operations are represented as template-like operators composed of rules that use information processing primitive functions and correlate CST relations with user summarization preferences. This paper builds on previous work in the area [1][2][4][5].

2 Content Selection Operators: Definition and Formalization

The general idea of this work is that, given a group of texts (annotated with CST) and some summarization preference (from the user), the content selection operators are applied in order to indicate the best information units to be part of the final summary. The existence of more than one operator mirrors the diverse interests that users may have: while some of them may be interested only in the main information, some may want to read general background information about the subject, while others might want to visualize contradictions among the information sources.

Formally, by content selection operators we mean computational artifacts that are able to process a content representation and to produce a condensed version of it. The operators we define take as input a "raw" initial rank of information units derived from the CST-annotated texts (which take the form of a graph – the CST graph) and

* This work was supported by FAPESP and CNPq.

T.A.S. Pardo et al. (Eds.): PROPOR 2010, LNAI 6001, pp. 25–29, 2010.

produces a refined rank, whose best ranked units (the "preferred ones" according to the specified user preference) should be selected for the final summary (respecting a specified compression rate). The CST relations from the CST graph are also included in the ranks, so that we do not need to keep consulting the graph for retrieving them.

The initial rank is simply built by taking all units from the CST graph and ordering them by the number of relations that they present. This process incorporates the underlying MDS assumption that the most important units are often repeated and elaborated in the several sources under consideration. Such units are usually highly connected with other ones by CST relations. Figure 1 shows a small hypothetical CST graph and the initial rank.

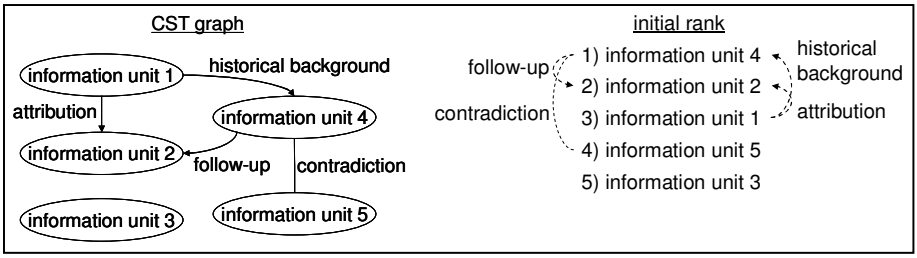

Fig. 1. Example of CST graph and initial rank

For the moment, when some units present the same number of relations, they are ranked in the order they are read from the graph. In this work, we consider sentences as our information units.

Our CST-based content selection operators are defined in a template-like format containing rules. The rules are specified as conditions and restrictions that must be satisfied for triggering actions, which are defined in terms of information processing primitive functions. Each rule is defined as follows:

$$\text{CONDITIONS, RESTRICTIONS} \Rightarrow \text{ACTIONS}$$

Each condition assumes the form CONDITION(S_i, S_j, Directionality, Relation) and is satisfied if there is the specified relation with the corresponding directionality among two sentences S_i and S_j (from S_i to S_j: →; the opposite way: ←; or no directionality: —). Restrictions are optional and may represent extra necessary requirements for the operator to be applied. If all the conditions and restrictions are satisfied, the actions are applied to the initial rank, producing its refined version. Actions are defined in terms of at least one of the three primitive functions below:

- MOVE_UP(S_i,S_j): re-ranks sentence j, putting it at a higher position in the rank, immediately after sentence i;
- SWITCH(S_i,S_j): switches the position of the sentences i and j in the rank;
- ELIMINATE(S_i): eliminates sentence i from the rank.

For this work, we defined and formalized 5 operators that represent possible user summarization preferences, namely, contextual information presentation, overview of evolving events, contradiction exhibition, authorship identification, and redundancy

treatment. We also see the process of building the initial rank as an operator, which gives preference for the main information. We refer to it by generic (or main information) operator. Each operator is defined by three fields: a reference name, its description, and the rules, which consist in its most important part. The first operator, the one for contextual information presentation, is shown in Figure 2. It looks for pairs of sentences (in any positions in the rank) that present the elaboration or the historical background CST relations (since these are the relations that give such contextual information) and move up the appropriate sentence in the rank. Therefore, the contextual information gets higher preference for being in the summary.

Name	Contextual information presentation
Description	Preference for historical and complementary information
Rules	CONDITION(S_i, S_j, ←, Elaboration) ⇒ MOVE_UP (S_i,S_j) CONDITION(S_i, S_j, ←, Historical background) ⇒ MOVE_UP(S_i,S_j)

Fig. 2. Contextual information presentation operator

The application of this operator to the initial rank of Figure 1 would produce the refined rank in Figure 3 (the initial rank is also shown for comparison). One may see that the historical information (information unit 1) move up in the rank, going right below the sentence to which it refers (information unit 4).

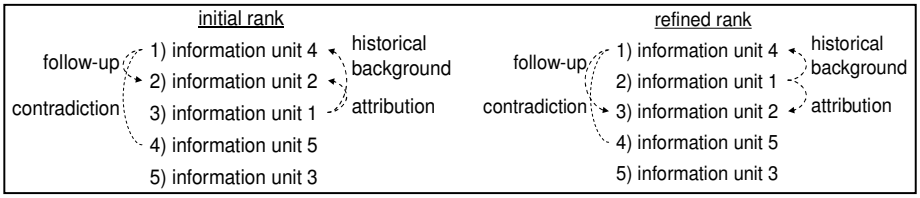

Fig. 3. Refined rank after application of the contextual information presentation operation

The next operator (Figure 4) is the one for giving preference for viewing the evolution of events in time. Such evolution is modeled in CST by the historical background and follow-up relations. Notice that, as the directionality does not matter, each rule appears twice, changing only the direction of the relation.

Name	Evolving events overview
Description	Preference for information about the evolution of events in time
Rules	CONDITION(S_i, S_j, ←, Historical background) ⇒ MOVE_UP(S_i,S_j) CONDITION(S_i, S_j, →, Historical background) ⇒ MOVE_UP(S_i,S_j) CONDITION(S_i, S_j, ←, Follow-up) ⇒ MOVE_UP(S_i,S_j) CONDITION(S_i, S_j, →, Follow-up) ⇒ MOVE_UP(S_i,S_j)

Fig. 4. Evolving events overview operator

Figure 5 shows the operator for exhibiting contradictions (which are expressed by the contradiction relation), while Figure 6 shows the operator for authorship identification (which is expressed by the attribution and citation relations).

Name	Contradiction exhibition
Description	Preference for contradictory information
Rules	CONDITION(S_i, S_j, —, Contradiction) $\Rightarrow$ MOVE_UP (S_i,S_j)

Fig. 5. Contradiction exhibition operator

Note that the rules in the authorship identification operator have more than one condition. All of them must be satisfied in order to the corresponding rule to be applied. This happens because the attribution and citation relations usually come with some other relation.

Name	Authorship identification
Description	Preference for authorship information
Rules	CONDITION(S_i, S_j, $\leftarrow$, Attribution), CONDITION(S_i, S_j, $\leftarrow$, Subsumption) $\Rightarrow$ SWITCH(S_i,S_j), ELIMINATE(S_i) CONDITION(S_i, S_j, $\leftarrow$, Citation), CONDITION(S_i, S_j, $\leftarrow$, Subsumption) $\Rightarrow$ SWITCH(S_i,S_j), ELIMINATE(S_i)

Fig. 6. Authorship identification operator

Finally, the redundancy treatment operator is shown in Figure 7. Besides the conditions, it also presents restrictions on the length of the sentences.

Name	Redundancy treatment
Description	Preference for non-redundant information
Rules	CONDITION(S_i, S_j, —, Identity) $\Rightarrow$ ELIMINATE(S_j) CONDITION(S_i, S_j, —, Equivalence), $\lvert S_i \rvert \leq \lvert S_j \rvert \Rightarrow$ ELIMINATE(S_j) CONDITION(S_i, S_j, —, Equivalence), $\lvert S_i \rvert > \lvert S_j \rvert \Rightarrow$ SWITCH(S_i,S_j), ELIMINATE(S_i) CONDITION(S_i, S_j, $\leftarrow$, Subsumption) $\Rightarrow$ SWITCH(S_i,S_j), ELIMINATE(S_i) CONDITION(S_i, S_j, $\rightarrow$, Subsumption) $\Rightarrow$ ELIMINATE(S_j)

Fig. 7. Redundancy treatment operator

It is important to say that there is another important CST relation that we are not dealing with at this moment: the overlap relation, which specifies that the sentences share some content, but also have unique parts. The overlap relation should be considered in the redundancy treatment and in the authorship identification operators at the cost of having some available automatic sentence fusion module. For instance, in the redundancy treatment operator, if there is an overlap relation among the sentences S_i and S_j, another sentence S_k should be built by fusing S_i and S_j.

An MDS system should be able to read the user preferences, select the appropriate operators, and produce the corresponding summary from a group of texts.

References

1. Aleixo, P., Pardo, T.A.S.: Finding Related Sentences in Multiple Documents for Multi-document Discourse Parsing of Brazilian Portuguese Texts. In: Anais do VI Workshop em Tecnologia da Informação e da Linguagem Humana – TIL, pp. 298–303 (2008)
2. Jorge, M.L.C., Pardo, T.A.S.: Content Selection Operators for Multidocument Summarization based on Cross-document Structure Theory. In: The Proceedings of the VII Brazilian Symposium in Information and Human Language Technology (2009)
3. Mani, I.: Automatic Summarization. John Benjamins Publishing Co., Amsterdam (2001)
4. Radev, D.R., McKeown, K.: Generating natural language summaries from multiple on-line sources. Computational Linguistics 24(3), 469–500 (1998)
5. Radev, D.R.: A common theory of information fusion from multiple text sources, step one: Cross-document structure. In: The Proceedings of the 1st ACL SIGDIAL Workshop on Discourse and Dialogue (2000)