Semantic Role Based Sentence Compression

Fatemeh Pourgholamali, Mohsen Kahani
Web Technology Lab, Faculty of Engineering
Ferdowsi University of Mashhad
Mashhad,Iran
Pourgholamali.ro_am@stu-mail.um.ac.ir , kahani@um.ac.ir

***Abstract*—In this paper, a new unsupervised sentence compression method is proposed. Sentences are tagged with Part Of Speech tags and semantic role labels. The proposed method relies on the semantic roles of sentences' parts. Moreover, in the process of compression, other sentences in the context are taken into account. The approach is applied in the context of multi-document summarization. Experiments showed better results than other state of the art approaches.**

Keywords- Sentence Compression; Part Of Speech; Semantic Role; Multi-Document Summarization; ROUGE

I. INTRODUCTION

Sentence compression is the process of removing some parts of a sentence while keeping its main information. Many applications such as text summarization and subtitle generation benefit from sentence compression. Many proposed approaches are tree-based [1], [2], [3], [4]. These methods compress sentences by making some changes in the parse tree of original sentences. Some approaches change the sentence directly to be compressed [5].

Most of introduced methods used supervised approaches [1], [5], [6]. These approaches require a robust and large training corpus that would take a lot of time and resources. To check the grammar of the output sentences, most approaches use a language model [1], [2], [7], and some methods apply hand-crafted rules [8]. Hand-crafted rules are not always general and applicable to any case. In addition, most introduced approaches treat sentences in the isolation and don't take into account other surrounding parts of the text.

We introduce a new unsupervised compression method, which is based on semantic roles of sentence elements. In addition, other related sentences in the context are taken into account in the sentence compression process. The results show that relying on the semantic levels resolves many grammatical challenges, and removes the need to use a language model or create and apply hand-crafted rules.

The proposed method has been used in the context of multi-document summarization. The evaluation results show improvements in regards to other approaches.

The structure of this paper is as follows. At first, the related works are discussed. In Section 3, the approach is explained in details. This is followed by the implementation and evaluation results and finally a conclusion is drawn.

II. RELATED WORKS

Most of the methods proposed for sentence compression are supervised and uses a language model to construct the compressed sentences and test the grammatically of them. Knight & Marcu [1] proposed two compression methods. One is based on the noisy channel concept and the other one on C4.5 [9]. First method uses language model $P(s)$ and a channel model $P(l|s)$ where s is the short (compressed) sentence and l is the original sentence. Best compression is the tree that maximizes $P(s)*P(l|s)$. To estimate $P(l|s)$, they used the probability of all the expansion operations, which would be needed to transform the parse tree of s into the parse tree of l.

Second method tries to transform l to the best s directly. This method learns when to delete and when to combine subtrees to achieve the goal. They extracted the training data from Ziff-Davis corpus, which contains articles about computer products. The extracted corpus was used for both algorithms' learning processes.

McDoland [5] has used the same corpus to learn weights and form vectors of weights. A scoring function that uses dot product of this vector with a vector of features extracted from the POS tags, n-grams and dependency trees, ranks each candidate tree. The sequence of words that maximizes the scoring function forms the best compression.

Berg et al. [10] proposed a jointly learning method to extract sentences and compress them within a unified model in the context of multi document summarization.

All supervised methods require training corpus to learn which parts can be omitted. Obtaining a robust training corpus is often time-consuming and difficult.

However, there are some unsupervised approaches, as well. Clark [8] proposed a method that finds the best compression using ILP [1]. The scoring function uses the language model to indicate which n-grams could be omitted with a high probability. To check the grammar of the output sentence, they apply hand-crafted constraints to the dependency tree of sentences.

Filippova proposed an unsupervised method for compressing dependency trees instead of source sentences [3]. The method transforms sentences into dependency trees and uses ILP to find the best compression. The objective function considers word significant scores and conditional probability of dependencies in the tree. Two kinds of constraints are applied to objective function, structural and

[1] Integer Linear Programming

syntactical. First constraint ensures that the preserved dependencies appears in a tree, and the second one checks if a node doesn't appear in the output, the dependent edges are omitted as well. The result tree should be linearized and transformed into a sentence.

Filippova also proposed a multi sentence compression approach in which related sentences form a graph [11]. The graph is constructed simply by adding words and matching similar ones in the same Part Of Speech (POS) tags. The shortest path in the graph, constructs the compressed sentence.

III. SEMANTIC ROLE BASED COMPRESSION

The proposed approach consists of three major phases as shown in Fig. 1 These phases are preprocessing, computing role similarity, and similarity based compression. Part of speech (POS) tagging and semantic role labeling (SRL) are two main preprocessing tasks in this method. Then, the similarities between sentences' semantic levels are computed, and finally, compressed sentences are produced. These steps are explained in more details, here.

A. Preprocessing

In this phase, the input sentences are tagged by POS and SRL tags. The Illinois University[2] tools have been used in this work, which have acceptable precisions.

In the process of semantic roles labeling task, each sentence is tagged with various semantic roles, such as subject, object, indirect object and some adjuncts, like adverbial modification and direction. Complex and long sentences can have two or even more semantic levels. Each level has its own semantic roles. A sample of this tool output for the following sentence is shown in Fig. 2.

"*A provincial official said that the water shortage caused the province's industrial output value to decrease by 3.6 billion yuan last year, and people in a number of cities and counties are short of drinking water supply*."

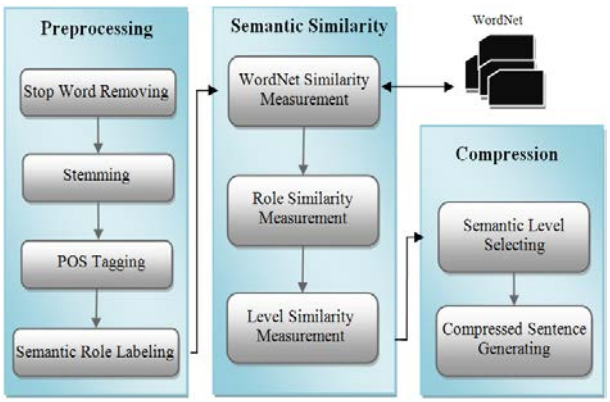

Figure 1. The architecture of proposed method

[2] http://cogcomp.cs.illinois.edu

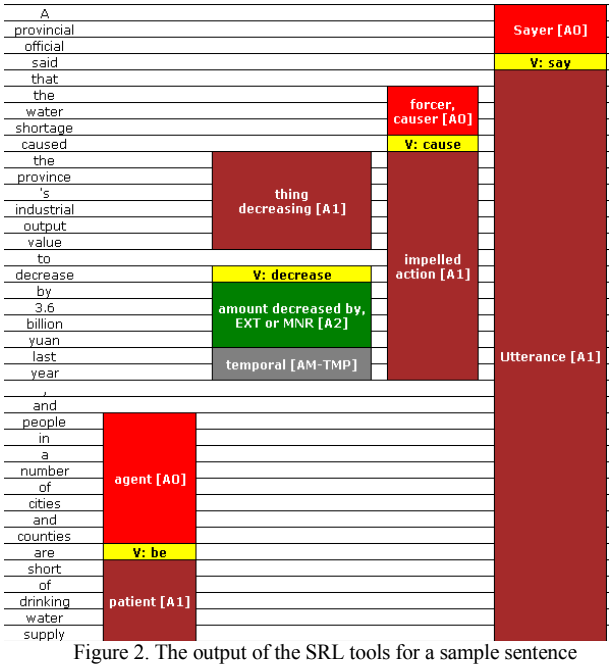

Figure 2. The output of the SRL tools for a sample sentence

A0, A1, and A2 labels denote subject, object, and indirect object roles respectively. Other labels in the brackets are adjunct roles.

POS tags are classified into four major categories, including nouns, verbs, adjectives, and adverbs. Also stemming and stopword removing are other preprocessing activities, which are applied at this step.

B. Semantic Level Similarity Metric

In this section, a metric for computing semantic similarity between levels of sentences is introduced. First, a metric for word similarity is needed, and it would be much better if semantic similarity is used. Therefore, we use the Lin WordNet similarity measure [12], which has obtained good results between proposed WordNet similarity measures in various evaluations [13]. To compute the semantic similarity between two words w_1 and w_2, Lin proposed the formula as follows:

$$sim_L(c_1,c_2) = \frac{2\log p(lso(c_1,c_2))}{\log p(c_1)+\log p(c_1)} \qquad (1)$$

where c_1 and c_2 are the most related pair of consepts in the taxonomy of WordNet that are sences of w_1 and w_2, $lso(c_1,c_2)$ is the most specialized common supreme between c_1 and c_2 in the taxonomy and $p(c)$ denotes the probability of encountering an instance of concept c.

In the following, computing similarities between levels are described. For the given sentences A,

$Roles\left(L_i^A\right) = \{r_1, r_2, \ldots, r_{n_A}\}$ is the set of roles in the ith semantic level of A,

$CommonRoles(L_i^A, L_j^B) = \{r_1, r_2, \ldots, r_{n_{cm}}\}$ is the common semantic roles in *i*th semantic level of *A* and *j*th semantic level of *B*,

$words_r\left(L_i^A\right) = \{w_1, w_2, \ldots, w_{m_A}\}$ is set of non-stopwords in the role *r* of sentence *A* after stemming,

Similarity of two equal roles is computed as formula (2):

$$RoleSim_r\left(words_r\left(L_i^A\right), words_r\left(L_j^B\right)\right) = \frac{\sum_{k=1}^{m_A}\left(\max_{w_B \in words_r\left(L_j^B\right), w_k \in words_r\left(L_i^A\right)} wordSim\left(w_k, w_B\right)\right)}{m_A + m_B} + \frac{\sum_{k=1}^{m_B}\left(\max_{w_A \in words_r\left(L_i^A\right), w_k \in words_r\left(L_j^B\right)} wordSim\left(w_A, w_k\right)\right)}{m_A + m_B} \quad (2)$$

The *wordSim* function is the Lin WordNet similarity measure described in the previous section. For each word that belongs to the word set of *A*, we sum its maximum similarity with words in the word set of *B*. this value is calculated for the word set of *B* and the final sum is normalized by dividing by the total number of the two word sets. In fact, the average of words similarities is calculated.

Now the similarity between two semantic levels should be computed. To do this, we use this formula:

$$LevelSim\left(L_i^A, L_j^B\right) = \frac{\sum_{r=1}^{n_{cm}} RoleSim_r\left(words_r\left(L_i^A\right), words_r\left(L_j^B\right)\right)}{n_{cm}} \quad (3)$$

The similarity between *i*th semantic level of sentence *A* and *j*th semantic level of sentence *B* is the sum of the similarity of the role pairs in the equal label that exists in *i*th level of sentence *A* and *j*th level of sentence *B*, normalized by the number of common roles between the two sentences. In the other word, the average of common roles similarities is computed.

C. Similarity Based Compression

In this phase, compression process is done via formula (4):

$$Compressed_k\left(A\right) = GenCompressed\left(A, l\right) \quad \text{where } LevelSim\left(L_l^A, L_j^{B_k}\right) \geq all\ LevelSim\left(L_i^A, L_j^{B_k}\right) \quad (4)$$

The similarity metric that was explained in the previous section, computed for any semantic level pair in two sentences. Given a sentence A, the similarity computations are applied to A and another sentence B_k in the discourse. The semantic level of A, which obtains the highest similarity, is selected and the core verb in the selected level with its existing arguments generates *k*th compressed form for the sentence A. this process is repeated for all sentences in the discourse. Since we customized this approach for multi-document summarization, a simple rule is applied: the core verb of the selected semantic level should not be a quotation verb.

The underlying idea in formula (4) comes from our studies in the multi-document summarization field. Since in the multi-document summarization there are a number of documents, and the goal is to extract most relative parts to the topic, which the documents are about, it sounds to be useful to utilize the similarities between the sentence parts and the surrounding context.

As a result, we get N compressed form for sentence A, with various compression rates. Note that the semantic roles are used and generating compressed sentence procedure is based on a semantic level and its arguments; output sentences have an acceptable grammar, and we do not need addition rules to check it. In addition, compressed sentences with various compression rates are created, and whichever is preferred and more appropriate can be chosen. In Table 1, some examples of original sentences and compressed form of them are shown.

IV. EVALUATION

We apply our method to sentences selected from the DUC corpus. Document Understanding Conferences (DUC) is run by the National Institute of Standards and Technology (NIST) and distributes standard data for automatic text summarization since 2001. DUC2007 dataset is the last and the most perfect one. Therefore, we used this dataset to evaluate our approach. This dataset contains 1125 documents and is developed for multi-document summarization purpose. Overall specification of this dataset is defined in Table II.

The evaluation results are compared with the summarization systems in DUC2007 and Filippova dependency tree based proposed method [3]. For summarization task, Lin [14] introduced some evaluation metrics with standard option as ROUGE[3] metrics. In this metric set, ROUGE-2 and ROUGE-SU4 have been given the best evaluating results and are used by researchers for the evaluation task. ROUGE-2 measure is based on bigrams shared between a system summary and human summaries. ROUGE-SU4 is based on both unigrams and skip-bigrams (separated bigrams by up to four words).

Our method is applied to the extracted sentences obtained from an extractive summarization method [15]. For evaluation, five random topics are selected, and ROUGE-2 and ROUGE-SU4 metrics are computed on the system's

[3] Recall-Oriented Understudy for Gisting Evaluation

output. Since various compressed forms for a sentence are obtained, five baselines (randomly) are defined based on which sentence would be selected as the final result.

The results shown in Fig. 3 and Fig. 4 indicate considerable improvement when compared to the average results of summarization systems in the DUC2007 dataset. In addition, our results show that selecting the longest compressed sentence (BaseLine.5) has made the best results. As Filippova [3] noted, the average recall and precision for sentence compression are calculated as the amount of grammatical relations shared between standard grammatical relations and system output ones, divided over the number of relations of human generated sentence and of system output respectively. The Relations are obtained by dependencies produced by Stanford Parser[4]. The results of the proposed method evaluation (in the two best base lines) as well as results reported by Filippova [3] are presented in Figure 5 and Fig. 6 the results show that in Base Line 5 (LONG) compressed sentences have shorter length and higher F-measure.

Base Line 1.Original sentence is used (no compression).
Base Line 2.Shortest sentence with lower bound to 35 words is selected. (LB.35)
Base Line 3.Shortest sentence with lower bound to 50 words is selected. (LB.50)
Base Line 4.Shortest sentence with lower bound to 80 words is selected. (LB.80)
Base Line 5.Longest sentence is selected. (LONG)

TABLE I. EXAMPLES OF SOME SENTENCES AND COMPRESSED FORMS OF THEM

sentence 1	Despite skepticism about the actual realization of a single European currency as scheduled on January 1, 1999, preparations for the design of the Euro note have already begun.
Sentence1 compressed	preparations for the design of the Euro note begun.
Sentence 2	Thailand is considering using the European single currency, the euro, in the country's foreign reserves, the Nation reported Tuesday.
Sentence2 compressed	Thailand considering using the European single currency, the euro.

TABLE II. OVERALL SPECIFICATION OF DUC2007 DATASET

# of Topics	45
# of Documents per Topics	25
# of Terms	531174
# of Terms without Stopwords & Stemmings	20057
# of Summarizer Systems	32
Evaluation methods	ROUGE 2 & ROUGE SU4

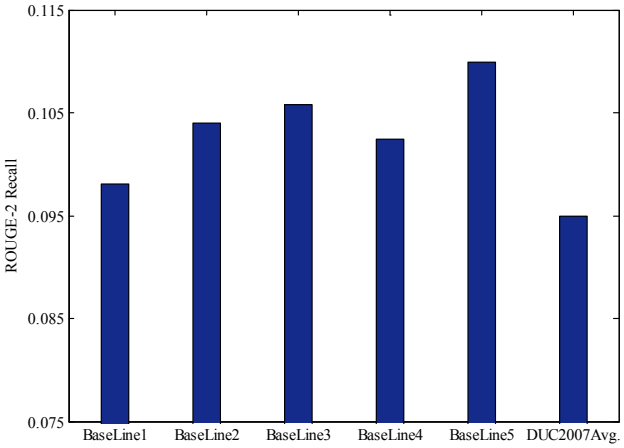

Figure 3. Results using ROUGE-2 metric

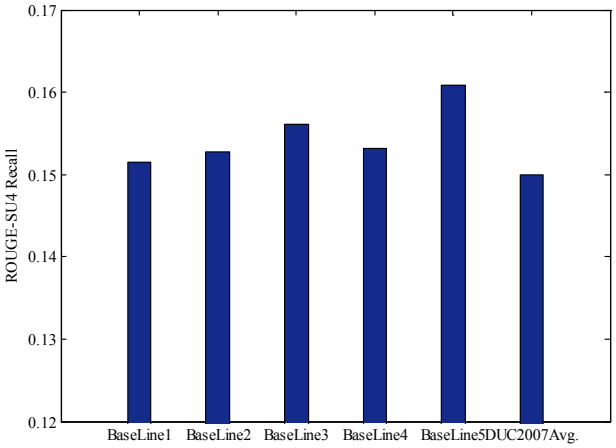

Figure 4. Results using ROUGE-SU4 metric

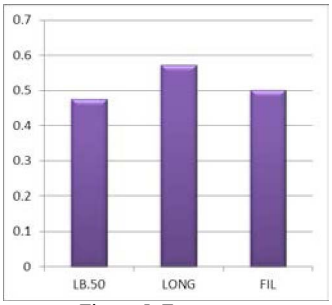

Figure 5. F-measure

[4] Available at http://nlp.stanford.edu

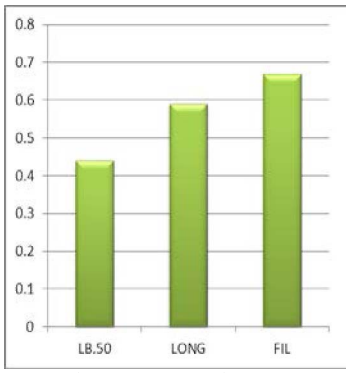

Figure 6. Compretion rate

V. CONCLUSION

In this paper, we proposed a new unsupervised approach based on semantic role labeling for the sentence compression task. The compressed sentences are produced by considering the semantic similarity between semantic levels of the sentence, which is going to be compressed, and some other sentences in the context. Relying on the semantic levels, produced sentences with acceptable grammar in most of the cases, and removes applying additional rules. The results show improvement in comparison to the state of the art approaches. In addition, our results show that the proposed method has been very effective for improving the results of automatic text summarization.

REFERENCES

[1] K. Knight and D. Marcu, "Summarization beyond sentence extraction: A probabilistic approach to sentence compression," Artificial Intelligence, 2002, pp. 91–107.

[2] M. Galley and K. McKeown, "Lexicalized markov grammars for sentence compression," In Proceedings of NAACL/HLT, 2007, pp. 180–187.

[3] K. Filippova and M. Strube, "Dependency tree based sentence compression," In ProceedingOf INLG-08, 2008, pp. 25–32.

[4] D. Galanis and I. Androutsopoulos, "An extractive supervised two-stage method for sentence compression," In Proceedings of NAACL, 2010.

[5] R. McDonald, "Discriminative sentence compression with soft syntactic evidence," In Proceedings of EACL, 2006, pp. 297–304.

[6] D. Galanis and I. Androutsopoulos, "A New Sentence Compression Dataset and Its Use in an Abstractive Generate-and-Rank Sentence Compressor." In Proceedings of the Language Generation and Evaluation Workshop (UCNLG+Eval), at the Conference on Empirical Methods on Natural Language Processing, Edinburgh, UK, 2011.

[7] J. Turner and E. Charniak, "Supervised and unsupervised learning for sentence compression," In Proceedings of ACL, 2005.

[8] Clarke and M. Lapata, "Global inference for sentence compression: An integer linear programming approach," Journal of Artificial Intelligence Research, 2008, pp. 399–429.

[9] J. R. Quinlan, "C4.5: Programs for Machine Learning," Morgan Kaufmann Publishers, 1993.

[10] T. Berg-Kirkpatrick, D. Gillick and D. Klein, "Jointly learning to extract and compress," In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1, HLT '11, Stroudsburg, PA, USA. Association for Computational Linguistics, 2011, pp. 481–490.

[11] K. Filippova, "Multi-Sentence Compression: Finding Shortest Path in Word Graphs," In Proceeding of the 23rd International Conference on Computational Linguistics, 2010, pp. 322-330.

[12] D. Lin, "An information-theoretic definition of similarity," In Proceedings of the International Conference on Machine Learning, Madison, 1998.

[13] A. Budanitsky and G. Hirst, "Evaluating wordnet-based measures of lexical semantic relatedness," Computational Linguistics, 2006, pp. 13–47.

[14] C.Y. Lin, "Rouge: A package for automatic evaluation of summaries," In Proceedings of the ACL-04Workshop: Text Summarization Branches Out, Barcelona, Spain, 2004, pp. 74–81.

[15] A. Pourmasoomi, M. Kahani, M. Kamyar and H. Kamyar, "Concept-Based Multi-Document Summarization," In Proceedings of 16th Annual National Conference of Computer Society of Iran (in Persian language), 2011, pp. 332-337.