One model to rule them all: ranking Slovene summarizers

Aleš Žagar and Marko Robnik-Šikonja

University of Ljubljana, Faculty of Computer and Information Science, Večna pot 113, 1000 Ljubljana, Slovenia {ales.zagar,marko.robnik}@fri.uni-lj.si

Abstract. Text summarization is an essential task in natural language processing, and researchers have developed various approaches over the years, ranging from rule-based systems to neural networks. However, there is no single model or approach that performs well on every type of text. We propose a system that recommends the most suitable summarization model for a given text. The proposed system employs a fully connected neural network that analyzes the input content and predicts which summarizer should score the best in terms of ROUGE score for a given input. The meta-model selects among four different summarization models, developed for the Slovene language, using different properties of the input, in particular its Doc2Vec document representation. The four Slovene summarization models deal with different challenges associated with text summarization in a less-resourced language. We evaluate the proposed SloMetaSum model performance automatically and parts of it manually. The results show that the system successfully automates the step of manually selecting the best model.

Keywords: Text summarization \cdot low-resource languages \cdot meta-model \cdot Slovene language.

1 Introduction

Text summarization identifies the essential information in a document or a collection of documents and presents it in a concise and coherent manner. In spite of the long efforts of natural language processing (NLP), text summarization is still a challenging task. With the explosive growth of digital information, summarizing large volumes of text into a shorter, more manageable form is becoming increasingly important.

There are two main approaches to text summarization: extractive and abstractive. Extractive summarization selects a subset of sentences or phrases from the original text that best represents the content. The selected sentences are combined to form a summary. Abstractive summarization, on the other hand, generates new sentences that capture the meaning of the original text. Extractive summarization is simpler and faster than abstractive summarization, but it can result in summaries that contain redundant and repetitive content. Abstractive

summarization is more challenging and requires more advanced natural language processing techniques, but it can produce human-like summaries.

State-of-the-art technology for text summarization has seen a significant shift in recent years with the rise of transformer neural network architectures, such as T5 [13] and GPT-3 [1]. This resulted in the summarization models whose summaries closely resemble those written by humans, with few repetitions and inaccuracies. These models are also capable of processing increasingly long content, enabling the creation of summaries for larger volumes of text. Consequently, state-of-the-art automatic summaries can be clear and easy to comprehend for end-users.

In the context of the less-resourced morphologically-rich Slovene language, text summarization is even more challenging than in English, due to limited availability of resources and data, as well as research. We produced four Slovene summarization models with different properties and trained them on different training data.¹. Our four models encompass two extraction summarizers (one based on a simple word frequency sentence selection, the other being graph-based), an abstractive T5-based model, and a hybrid extractive-abstractive model. In general, the T5-based transformer model works best but may not generalize well for all types of input text. Therefore, we address the problem of which summarization model is the most appropriate for a given text, based on text length and genre.

We propose a novel Slovene summarization system (named SloMetaSum), consisting of extractive, abstractive, and hybrid summarizers and a meta-model that selects among them. The proposed meta-system consists of a fully connected neural network that analyzes the input content and recommends the most suitable summarization model for a given text. To achieve this, SloMetaSum uses the Doc2Vec [7] numerical representation of documents and predicts the ROUGE scores for each of the summarizers. By using a combination of approaches, the system can effectively generate high-quality summaries that are informative and easy to understand for many types of text, regardless of their length and genre.

Our contributions are:

- We have developed four summarization models that can effectively summarize text of varying lengths and genres, making them versatile for a range of applications.
- We overcame the challenges of the low-resourced Slovene language, and created high-performing models for summarizing Slovene text.
- We have also created a meta-model that can recommend the best-suited summarization model for a given text based on factors such as length, complexity, level of abstraction, and intended use case.

¹ Within the scope of the RSDO project: https://www.cjvt.si/rsdo/

² The demo is available at https://slovenscina.eu/en/povzemanje. The code repositories are available at https://github.com/azagsam/metamodel and https://github.com/clarinsi/SloSummarizer.

The rest of the paper is organized as follows. We present related research in Section 2. Section 3 describes the datasets. In Section 4, we describe summarization systems and the meta-model. We present our experiments and discuss the findings in Section 5. Section 6 concludes and recommends future research.

2 Related work

Early approaches to text summarization relied on statistical frequencies of words, sentence position, and sentences containing keywords [12]. These approaches aimed to extract important sentences or phrases from a text and generate a summary by concatenating them. Abstractive methods involved deleting less important words from the text to create a summary [6].

Graph-based methods have been another popular approach to text summarization. In this approach, the document is represented as a graph, where sentences are nodes, and edges represent the relationships between them. The graph is then used to generate a summary by selecting the most important sentences. This method has been explored in several works [10], [3].

With the advent of neural networks, there has been an increasing interest in developing abstractive summarization techniques. Early neural abstractive systems used methods such as LSTM and other recurrent neural networks [14], [11]. However, transformer-based architectures have emerged as state-of-the-art models for abstractive text summarization [18], [9]. These models use self-attention mechanisms to selectively focus on important parts of the text and can generate more fluent and coherent summaries compared to earlier methods.

While several approaches have been proposed for text summarization, many of them are designed to handle specific genres or types of text. In this work, our goal is to build a summarization system that can handle every type of text and genre with every possible property that can appear in the real world. This includes texts of varying lengths, topics, styles, and summaries that capture the most important information in the text. Achieving this goal requires developing a robust and adaptable model that can learn to summarize texts of diverse types and produce high-quality summaries.

3 Datasets

In this section, we describe the datasets we used in our research. Below, we provide a short description of the datasets, with their statistics contained in Table 1.

The STA dataset (general news articles from the Slovenian Press Agency) consists of 366,126 documents and the first paragraph of each article was used as a proxy for summary since the dataset does not contain hand-written human summaries. This is a common technique in text summarization, especially in languages that do not have dedicated news article summarization datasets such as English.

AutoSentiNews [2] is a similar dataset to STA, consisting of 256,567 articles from the Slovenian news portals 24ur, Dnevnik, Finance, RTVSlo, and Žurnal24. The summaries are produced from the first paragraph in the same way as they are in the STA dataset.

The SURS dataset is a small financial news dataset from the Slovenian statistical office and consists of 4,073 documents.

The KAS corpus of Slovene academic writing [16] consists of BSc/BA, MSc/MA, and PhD theses written from 2000 - 2018 and gathered from the digital libraries of Slovene higher education institutions via the Slovene Open Science portal ³. The corpus contains human-written abstracts of academic texts.

CNN/Daily Mail dataset [5] is for text summarization. It has human-generated abstractive summary bullets from news stories on CNN and Daily Mail websites. The corpus has 286,817 training pairs, 13,368 validation pairs, and 11,487 test pairs. The source documents have 766 words and the summaries consist of 53 words on average. We translated the dataset in Slovene using machine translation [8].

Dataset	Number of documents
STA	334,696
AutoSentiNews	256,567
SURS	4,073
KAS	82,308
Total	$677,\!644$

Table 1: Corpora and datasets used to train a Doc2vec document representation model and the meta-model.

4 The summarization models and the meta-model

In this section, we describe the components of our SloMetaSum system which consists of four summarization models, a technique for document representation, and the meta-model.

4.1 Summarization models

We produced four summarization models, described below.

Sumbasic [12] uses a simple word frequency approach to select the most informative sentences. The **graph-based** summarization model [17] was inspired by the TextRank algorithm [10] and uses centrality scores of sentences to rank them. Both models belong to extractive methods and can be used on documents

³ http://openscience.si/

of any size. In contrast to the original TextRank, we used the transformer-based LaBSE sentence encoder [4], to numerically represent sentences. The **T5-article** abstractive summarization model uses a pre-trained Slovene T5 model [15] and is fine-tuned on a machine-translated CNN/Daily Mail dataset [5] using the Slovene machine translation system [8]. The **hybrid-long** summarization model is a combination of the graph-based and the T5-article model. It first constructs a short text by concatenating the most informative sentences (extractive step). In the next, abstractive step, these sentences are summarized with the T5-article summarizer.

4.2 Doc2Vec model representation

To select the most suitable summarization method for a given text, the metamodel has to get information about different text properties. We apply the Doc2Vec model for document representation and train it on the Slovene documents presented in Table 1 (without abstracts). In the preprocessing step, we removed high-frequency words that do not contribute to the meaning of a document, such as pronouns, conjunctions, etc.; to further reduce the number of different words, we lemmatized the whole dataset.

4.3 Meta-model

Our meta-model consists of a fully connected neural network, trained to predict the ROUGE scores of the summarizers. For a training dataset, we randomly selected 93.419 examples from the raw concatenated dataset. After that, each of our four summarizers produced a summary for all examples. We calculated ROUGE scores between the reference and generated summaries. ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is a metric most commonly used for the evaluation of automatically generated text summaries. It measures the quality of a summary by the number of overlapping units (n-grams, sequences of texts, etc.) between summaries created by humans and summaries created by summarization systems. ROUGE is not a single metric but a family of metrics. The most commonly used are ROUGE-N and ROUGE-L. The first measures the overlapping of n-grams (typically unigrams and bigrams), while the second measures the longest common subsequence found in both summaries. As an input to our meta-model, we use four ROUGE F1-scores (ROUGE-1, ROUGE-2, ROUGE-L, ROUGE-LSum) that show how good the generated summaries are. We split data into train, validation, and test sets in ratios of 90:5:5.

The sizes of both datasets are presented in Table 2. In Table 3, we present the average ROUGE values of our summarizers on long and short texts. Summarizers that are specialized for short texts achieve better results on short texts and vice versa.

Model	Training size
Doc2Vec	677,644
Meta-model	$93,\!419$

Table 2: Number of training samples for each model.

	t5-article	$\operatorname{sumbasic}$	graph-based	hybrid-long
Short	14,01	$13,\!11$	$13,\!15$	12,55
Long	$10,\!51$	$13,\!12$	17,71	17,59

Table 3: Summarizers ROUGE scores for long and short texts. The best scores for short and long texts are in bold.

5 Results

In this section, we present our results and evaluation. We report the performance of the Doc2Vec model and Meta-model in each separate subsection.

5.1 Doc2Vec

We used the following hyperparameters for training the Doc2Vec document representation model: the maximum allowed vocabulary size is 100,000, the size of the vector used for word representation is 256, the window size of context words is 5, the minimum frequency of a word to be included in the vocabulary is 1, and the total number of epochs or iterations for training the model is 5.

We evaluated the Doc2Vec model using manual and automatic techniques. For manual analysis, we inspected the top 3 most similar returned documents for each of a few randomly chosen samples using the cosine similarity and observe whether the topics of the documents overlap. The topics of the documents were similar in most cases and based on that we concluded that the model works as expected. The automatic evaluation was part of the whole pipeline, where the model hyperparameters were tuned to optimize the loss of the meta-model.

5.2 Meta-model

Our final results are presented in Table 5. We compared the proposed meta-model selection mechanism with three baselines. The *Mean-baseline* model simply takes the predictions for each summarization model and averages them. The highest-scoring model is always selected. The *Tree* uses a regression tree; using the hyperparameter grid search, the minimum number of samples required to split an internal node is 100. The *Forest* method uses a random forest; we experimented with similar values as for the Tree model and set the number of tree estimators to 300.

Our best model is a neural network with two hidden layers. The hidden layers contain 1024 neurons, and we used a validation split of 0.1 during the training process. The activation function used for this model is the rectified linear unit (ReLU). In addition, for the early stopping scheduling strategy, we set the patience parameter to 2. The loss function utilized for this model is the mean squared error.

Meta-model stopped learning after 7 epochs and performed almost 15 points above Mean-baseline on the test set. We observed that choosing different hyperparameters does not seem to significantly affect the results. We experimented with different hidden layer sizes, numbers of units, and activation functions. We also tried different max vocabulary and window sizes of the Doc2Vec model. We report only the values of the best model.

Overall, this model was found to be the most effective among the meta-model selection strategies we tested. The high number of neurons in the hidden layer likely contributed to its superior performance, as it allows for a greater degree of complexity in the model's representation of the data.

We further experimented with two variations of the meta-model. Meta-modellength adds another input neuron that explicitly encodes the input length. We found that this does not improve the model and hypothesize that academic texts are of different genres and the document embedding technique covers it well already. We also tried to balance data since the original dataset contains a 1:5 ratio of long to short texts which rises a potential issue of overfitting on short texts. We reduced the number of short texts in a training set to get a balanced dataset of 16,932 samples for our Meta-model-balanced model. This resulted in a worse-performing model but still better than Mean-baseline.

Table 4 shows the frequencies of how many times each model was recommended by a meta-model out of 1000 samples from a test set. We can see that the t5-article model was recommended the most, with a count of 595 out of 1000 samples. The hybrid-long model was recommended 254 times, followed by the Sumbasic model, which was recommended 80 times. The graph-based model was recommended the least, with a count of 71 out of 1000 samples.

Model	Count
t5-article	595
hybrid-long	254
sumbasic	80
graph-based	71
Total	1000

Table 4: Frequencies of how many times each model was recommended by the meta-model out of 1000 samples from the test set.

According to Table 6, the graph-based method achieved the highest F1-score of 0.48, with a precision of 0.38 and recall of 0.67. The hybrid-long method

Model	Mean squared error
Mean-baseline	84.493
Tree	81.631
Random forest	74.975
Meta-model-baseline	70.066
Meta-model-length	70.146
Meta-model-balanced	79.044

Table 5: Results of our four models on the test set. Meta-model-baseline showed significant improvement over Mean-baseline and tree methods. Encoding the length feature explicitly or balancing the dataset did not improve the results.

Method	Precision	Recall	F1-score	Support
t5-article	0.33	0.11	0.16	1069
hybrid-long	0.25	0.34	0.29	817
$\operatorname{sumbasic}$	0.28	0.10	0.15	1196
graph-based	0.38	0.67	0.48	1589

Table 6: Classification report. The table includes precision, recall, and F1-score for each method, as well as the number of instances in the test set (Support). The methods include t5-article, hybrid-long, sumbasic, and graph-based. Test accuracy was 0.34.

achieved F1-score of 0.29, with precision 0.25 and recall 0.34. The sumbasic method produced F1-score of 0.15, precision 0.28, and recall 0.10. Finally, the t5-article method achieved the lowest F1-score of 0.16, with precision of 0.33 and recall of 0.11. Overall, the test accuracy for all methods combined was 0.34.

5.3 Meta-model vs. the rest

In Table 7, we present the final evaluation results obtained from our experiments on the test set. It is noteworthy that the proposed Meta-model outperformed all other models across all ROUGE scores. This result highlights the effectiveness and superiority of the Meta-model in selecting the most suitable summarization approach for a given text. This outcome showcases the potential of our approach in automating the process of selecting the best summarization model, eliminating the need for manual intervention.

6 Conclusion

In this paper, we proposed a novel system for extractive, abstractive, and hybrid summarization tasks. Our system consists of a trained fully connected neural network that analyzes the input content and recommends the most suitable summarization model for a given text. This approach addresses the problem of

Model	ROUGE-1	ROUGE-2	ROUGE-L
t5-article	19.01	5.61	13.52
graph-based	19.47	5.52	12.50
hybrid-long	18.55	5.42	11.73
sumbasic	18.86	5.04	12.25
Meta-model	20.38	5.85	13.67

Table 7: Performance on the test set for all models. Meta-model achieves the best results in all three categories.

selecting the appropriate model for a new text, which can be short, long, and of various genres, and can come from almost anywhere when used in production. Our system provides a more effective and efficient way of generating high-quality summaries for Slovene texts.

While the proposed SloMetaSum model presents an innovative solution to the problem of selecting the most suitable summarization model for a given text, it is not without its weaknesses. One major drawback is the reliance on the ROUGE score as the sole criterion for model selection. While ROUGE is a commonly used metric in the field of text summarization, it does not always accurately reflect the quality of a summary or capture its coherence and readability. Another potential weakness is the limited scope of the study, which focuses exclusively on the Slovene language. While the four summarization models developed for Slovene are an important contribution to the field, they may not generalize well to other less-resourced languages since it requieres a good automatic translation system.

Future work could involve extending this system to other languages. Another area for future work could involve comparing the proposed system with recent large language models. In addition to evaluating the technical performance of the system, it would also be useful to conduct user studies to assess its usefulness and effectiveness in real-world scenarios. For example, researchers could design experiments to evaluate the system's ability to summarize news articles, academic papers, and other types of content that people encounter in their daily lives.

Acknowledgments

The work was partially supported by the Slovenian Research Agency (ARRS) core research programme P6-0411, as well as projects J6-2581, J7-3159, and CRP V5-2297.

References

 Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. arXiv preprint arXiv:2005.14165 (2020)

- 10 A. Žagar & M. Robnik-Šikonja
- Bučar, J.: Automatically sentiment annotated slovenian news corpus AutoSentiNews 1.0 (2017), http://hdl.handle.net/11356/1109, slovenian language resource repository CLARIN.SI
- 3. Erkan, G., Radev, D.R.: Lexrank: Graph-based lexical centrality as salience in text summarization. Journal of artificial intelligence research 22, 457–479 (2004)
- Feng, F., Yang, Y., Cer, D., Arivazhagan, N., Wang, W.: Language-agnostic BERT sentence embedding. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 878–891 (2022)
- Hermann, K.M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., Blunsom, P.: Teaching machines to read and comprehend. Advances in neural information processing systems 28 (2015)
- Knight, K., Marcu, D.: Summarization beyond sentence extraction: A probabilistic approach to sentence compression. Artificial Intelligence 139(1), 91–107 (2002)
- Le, Q., Mikolov, T.: Distributed representations of sentences and documents. In: International conference on machine learning. pp. 1188–1196. PMLR (2014)
- Lebar Bajec, I., Repar, A., Bajec, M., Bajec, Ž., Rizvič, M.: NeMo neural machine translation service RSDO-DS4-NMT-API 1.0 (2022), http://hdl.handle.net/11356/1739, slovenian language resource repository CLARIN.SI
- Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L.: BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 7871–7880 (2020)
- Mihalcea, R., Tarau, P.: Textrank: Bringing order into text. In: Proceedings of the 2004 conference on empirical methods in natural language processing. pp. 404–411 (2004)
- Nallapati, R., Zhou, B., dos Santos, C., Gulçehre, Ç., Xiang, B.: Abstractive text summarization using sequence-to-sequence rnns and beyond. In: Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning. pp. 280–290 (2016)
- 12. Nenkova, A., Vanderwende, L.: The impact of frequency on summarization. Tech. rep., Microsoft Research (2005)
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research 21(140), 1–67 (2020)
- See, A., Liu, P.J., Manning, C.D.: Get to the point: Summarization with pointergenerator networks. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 1073–1083 (2017)
- Ulčar, M., Robnik-Šikonja, M.: Sequence to sequence pretraining for a lessresourced slovenian language. arXiv preprint arXiv:2207.13988 (2022)
- 16. Žagar, A., Kavaš, M., Robnik-Šikonja, M., Erjavec, T., Fišer, D., Ljubešić, N., Ferme, M., Borovič, M., Boškovič, B., Ojsteršek, M., Hrovat, G.: Corpus of academic slovene KAS 2.0 (2022), http://hdl.handle.net/11356/1448, slovenian language resource repository CLARIN.SI
- Žagar, A., Robnik-Šikonja, M.: Unsupervised approach to multilingual user comments summarization. In: Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation. pp. 89–98. Association for Computational Linguistics (Apr 2021)

 Zhang, J., Zhao, Y., Saleh, M., Liu, P.: Pegasus: Pre-training with extracted gapsentences for abstractive summarization. In: International Conference on Machine Learning. pp. 11328–11339. PMLR (2020)