Text Summarization Methods and Applications in Romanian NLP

Bratfalean Dragos
Artificial Intelligence *and Signal Processing, Technical University of Cluj-Napoca*

Abstract— **Text summarization is a key task in Natural Language Processing (NLP) that aims to produce a concise and informative version of a longer text. While extensive research exists for high-resource languages such as English, summarization for Romanian remains comparatively underexplored. This document presents an overview of text summarization methods and discusses their applicability, challenges, and real-world use cases in Romanian NLP.**

Keywords — ***NLP, text summarization, Romanian***

I. Introduction (*Heading 1*)

The exponential growth of digital textual data generated through news platforms, social media, governmental portals, academic publications, and enterprise systems has made it increasingly difficult for users to efficiently access and process information. As a result, automatic text summarization has emerged as a crucial task in Natural Language Processing (NLP), aiming to reduce information overload by producing concise representations of longer documents while preserving their most important content.

Text summarization systems are now widely integrated into real-world applications such as search engines, news aggregators, digital assistants, and decision-support systems. These systems help users save time, improve comprehension, and support faster decision-making. Depending on the application domain, summaries may vary in length, level of detail, and degree of abstraction.

In the context of Romanian language processing, text summarization presents both significant opportunities and notable challenges. Romanian is considered a medium- to low-resource language in NLP, with fewer annotated datasets, pretrained models, and linguistic tools compared to high-resource languages such as English. Additionally, Romanian's rich morphology, inflectional complexity, and relatively flexible word order increase the difficulty of accurately identifying salient information and generating coherent summaries.

Despite these challenges, recent advances in multilingual and transformer-based models have created new possibilities for Romanian text summarization. By leveraging transfer learning, cross-lingual representations, and multilingual pretrained architectures, it is now possible to build effective summarization systems even in the absence of large, language-specific datasets.

II. Text Summarization Methods

Text summarization techniques are generally divided into extractive and abstract approaches.

A. Extractive Summarization

Extractive summarization frames the summarization task as the selection of a subset of sentences from the source document that best represents its main content. The generated summary consists entirely of sentences taken verbatim from the original text, without introducing new linguistic material.

Early extractive approaches rely on statistical and heuristic features to estimate sentence importance, including TF–IDF weights, sentence position, length, and the presence of salient terms or named entities. Graph-based methods such as TextRank and LexRank further model documents as sentence similarity graphs and apply ranking algorithms to identify central sentences.

More recent extractive systems adopt supervised and neural techniques, treating sentence selection as a classification or ranking problem. Transformer-based encoders are commonly used to produce contextualized sentence representations, which are then scored using attention mechanisms or learned classifiers.

Extractive summarization is particularly well suited to low-resource languages such as Romanian, as it requires limited annotated data and preserves grammaticality and factual consistency. However, its inability to paraphrase or merge information often leads to summaries that are less coherent or overly detailed. Despite these limitations, extractive methods remain a strong and reliable baseline for Romanian NLP applications.

.

B. Abstractive Summarization

Abstractive summarization aims to generate concise summaries that paraphrase and reorganize the content of the source document, rather than extracting sentences verbatim. This approach models summarization as a natural language generation task and seeks to capture the underlying meaning of the text while expressing it in new linguistic forms.

Early abstractive methods are based on sequence-tosequence architectures with attention mechanisms, which map an input document to a shorter output sequence. More recent approaches predominantly rely on transformer-based models, such as BART, T5, and PEGASUS, which have demonstrated strong performance by leveraging large-scale pretraining on summarization and language modelling objectives.

In practice, abstractive summarization offers more coherent and compact summaries than extractive methods, but it also introduces challenges related to data requirements and factual consistency. For Romanian, abstractive approaches are typically implemented using multilingual pretrained models or transfer learning techniques, due to the scarcity of large, language-specific summarization datasets.

Despite these limitations, abstractive summarization represents an important direction for advancing Romanian NLP systems.

III. Approaches for Romanian NLP

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before formatting. Please note sections A-D below for more information on proofreading, spelling and grammar.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads-the template will do that for you.

A. Traditional and Extractive Approaches

Due to limited datasets, extractive methods are widely used for Romanian. These methods often rely on:

- TF–IDF scoring
- Sentence position heuristics
- Romanian POS tagging and lemmatization

B. Neural and Multilingual Models

Multilingual transformer models provide practical solutions for Romanian summarization:

- mBART-50
- mT5
- XLM-R

These models can be applied in zero-shot settings or fine-tuned using translated dataset.

C. Transfer Learning Strategies

Transfer learning is essential for Romanian text summarization due to the scarcity of large, annotated datasets. Most approaches leverage multilingual pretrained models, such as mBART, mT5, or XLM-R, which learn shared crosslingual representations and can be applied to Romanian in zero-shot or lightly fine-tuned settings.

A common strategy is dataset transfer via machine translation, where large English summarization corpora are translated into Romanian for supervised fine-tuning. Alternatively, Romanian texts can be translated into English, summarized using high-performing English models, and then translated back. While effective, these pipelines may introduce translation noise and error propagation.

Cross-lingual fine-tuning further improves performance by adapting models pretrained or fine-tuned on high-resource languages to Romanian using smaller, domain-specific datasets. In addition, parameter-efficient fine-tuning methods, such as adapter layers, reduce computational cost while maintaining competitive performance. Overall, transfer learning remains a key enabler for practical Romanian summarization systems.

Evaluating the quality of automatically generated summaries is a non-trivial task, as it involves both objective and subjective criteria. In Romanian text summarization, evaluation is further complicated by limited reference summaries and the scarcity of standardized benchmarks.

- The word “data” is plural, not singular.
- The subscript for the permeability of vacuum μ_0, and other common scientific constants, is zero with subscript formatting, not a lowercase letter “o”.
- In American English, commas, semicolons, periods, question and exclamation marks are located within quotation marks only when a complete thought or name is cited, such as a title or full quotation. When quotation marks are used, instead of a bold or italic typeface, to highlight a word or phrase, punctuation should appear outside of the quotation marks. A parenthetical phrase or statement at the end of a sentence is punctuated outside of the closing parenthesis (like this). (A parenthetical sentence is punctuated within the parentheses.)
- A graph within a graph is an “inset”, not an “insert”. The word alternatively is preferred to the word “alternately” (unless you really mean something that alternates).
- Do not use the word “essentially” to mean “approximately” or “effectively”.
- In your paper title, if the words “that uses” can accurately replace the word “using”, capitalize the “u”; if not, keep using lower-cased.
- Be aware of the different meanings of the homophones “affect” and “effect”, “complement” and “compliment”, “discreet” and “discrete”, “principal” and “principle”.
- Do not confuse “imply” and “infer”.
- The prefix “non” is not a word; it should be joined to the word it modifies, usually without a hyphen.
- There is no period after the “et” in the Latin abbreviation “et al.”.

- The abbreviation “i.e.” means “that is”, and the abbreviation “e.g.” means “for example”.

An excellent style manual for science writers is [7].

IV. Evaluation methods

Evaluating the quality of automatically generated summaries is a non-trivial task, as it involves both objective and subjective criteria. In Romanian text summarization, evaluation is further complicated by limited reference summaries and the scarcity of standardized benchmarks.

A. Automatic Evaluation Metrics

Automatic metrics are widely used due to their scalability and reproducibility. The most common family of metrics is ROUGE (Recall-Oriented Understudy for Gisting Evaluation), including ROUGE-1, ROUGE-2 as in (1) and ROUGE-L as in (2), which measure lexical overlap between system-generated summaries and reference summaries. Despite their simplicity, ROUGE metrics remain a standard baseline in summarization research, including for Romanian.

$$ROUGE - N = \frac{Number\ of\ matching\ n-grams}{Total\ n-grams\ in\ the\ reference} \quad (1)$$

$$ROUGE - L = \frac{(1+\beta^2)\cdot P\cdot R}{Total\ n-grams\ in\ the\ reference} \quad (2)$$

$$where\ P = precision, R = recall, \beta\ is\ typically\ set\ to\ 1$$

Other metrics, such as BLEU, are occasionally employed but are generally less suited for summarization due to their focus on precision rather than recall. More recent semantic based metrics, including BERT Score and other embedding-based similarity measures, are particularly relevant for Romanian, as they can leverage multilingual language models to capture semantic equivalence beyond surface-level word overlap.

B. Human Evaluation

Human evaluation remains essential for assessing aspects of summary quality that are difficult to capture automatically. Common evaluation criteria include informativeness (coverage of key content), fluency (grammaticality and readability), coherence, and faithfulness to the source document. Human judgments are especially important for abstractive summarization, where models may generate fluent but factually incorrect statements

C. Evaluation Challenges for Romanian

For Romanian, evaluation is often constrained by the lack of multiple high-quality reference summaries and standardized test sets. As a result, many studies rely on smallscale human evaluations or translated reference summaries, which may introduce bias. Developing reliable evaluation protocols and benchmark datasets for Romanian summarization remains an open research challenge.

VI. ARCHITECURE AND OBJECTIVE

A. BART Architecture

BART (Bidirectional and Auto-Regressive Transformers) is a sequence-to-sequence model based on the standard Transformer encoder–decoder architecture. Both the encoder and decoder are implemented as stacks of Transformer blocks composed of multi-head self-attention, position-wise feed-forward networks, residual connections, and layer normalization.

In its original formulation, BART is pre-trained on English-only corpora and typically uses a deep configuration with symmetric encoder and decoder stacks (e.g., 12 layers each). The model operates on subword units produced by byte-pair encoding or SentencePiece tokenization. Positional embeddings are added to token embeddings to encode word order. The decoder is autoregressive and attends both to previously generated tokens (via masked self-attention) and to the encoder representations (via cross-attention).

The architecture itself is task-agnostic and does not include any language-specific components, making BART primarily suitable for monolingual generation and understanding tasks.

B. mBART Architecture

mBART extends the BART architecture to the multilingual setting while retaining the same core Transformer encoder–decoder design. Architecturally, mBART uses a **single shared model** across all languages, with no language-specific encoders or decoders. A typical mBART configuration consists of **12 encoder layers and 12 decoder layers**, a model dimension of **1024**, and **16 attention heads**, resulting in approximately **680 million parameters**.

A key architectural addition in mBART is the use of a **language identifier token (<LID>)**. This token is appended to the encoder input and used as the initial token for the decoder, explicitly conditioning the model on the source or target language. This mechanism enables multilingual generation and cross-lingual transfer without modifying the underlying architecture.

mBART uses a large shared subword vocabulary (e.g., 250k SentencePiece tokens) learned from multilingual corpora, allowing it to generalize across languages with different scripts and morphological properties. An additional layer normalization is applied on top of both the encoder and decoder stacks to improve training stability, particularly under mixed-precision (FP16) training.

C. Multilingual Denoising in mBART

While BART applies the denoising objective to English-only data, mBART generalizes this objective to **multilingual monolingual corpora**. During pretraining, each training instance is sampled from a particular language corpus, corrupted using the same noise functions, and reconstructed by the shared model conditioned on the corresponding language ID token.

By pretraining on full documents across multiple languages, mBART learns language-agnostic representations at higher layers while retaining language-specific lexical knowledge in embeddings. Importantly, the model is pre-trained as a complete sequence-to-sequence system, allowing it to be

fine-tuned directly for supervised, document-level, or unsupervised machine translation without architectural changes.

V. PRACTICAL APPLICATION

mBART **(Multilingual BART)** is a **sequence-tosequence Transformer model** designed for multilingual text generation tasks. It follows the standard **encoder-decoder architecture**:

- **Encoder:** Processes the input text into contextual representations (BERT).
- **Decoder:** Autoregressively generates output sequences, conditioned on the encoder's representations, (GPT).
- The model uses **multi-head self-attention**, **layer normalization**, and **residual connections**, consistent with the Transformer architecture.
- mBART-large-50 supports **50 languages**, sharing parameters across languages to enable cross-lingual transfer.

For pretraining, mBART employs a **denoising autoencoding objective**:

1. **Text corruption:** Input text is corrupted by masking spans, shuffling sentences, or replacing spans with special tokens.
2. **Sequence reconstruction:** The model is trained to reconstruct the original text from the corrupted version.

- Specifically, spans of text are replaced with **[MASK] tokens** (text infilling), and the model predicts the missing tokens.
- This trains the model to capture both **content and structure** of the language, as well as cross-lingual patterns.

By learning to recover corrupted text across multiple languages, mBART acquires **strong multilingual contextual representations** that transfer effectively to downstream tasks such as summarization, translation, and text generation.

The following table shows the result of evaluating ROUGE metric for the result of summarization and a reference.

"România este un stat situat în sud-estul Europei Centrale, pe cursul inferior al Dunării, la nord de peninsula Balcanică și la țărmul nord-vestic al Mării Negre. Numele României derivă din cuvântul latin Romanus, care înseamnă „cetățean al Romei"; regiunea a fost un avanpost al Imperiului Roman în secolul al II-lea d.Hr. Acest nume a fost adoptat în 1861, la doi ani după Unirea Principatelor Române, alegerea sa având și rolul de a sublinia moștenirea comună de origine latină a celor trei mari regiuni istorice; Țara Românească, Moldova și Transilvania, în contextul procesului lor treptat de unificare, desfășurat între mijlocul secolului al XIX-lea și începutul secolului al XX-lea".

TABLE I. ROUGE RESULTS TABLE

Table Head	**Table Column Head**		
	Precision	***Recall***	***F1-score***
ROUGE-1	0.4286	0.3061	0.3571
ROUGE-2	0.1765	0.1250	0.1463
ROUGEL	0.4000	0.2857	0.3333

REFERENCES

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2020). *BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension*. Proceedings of ACL.

Liu, Y., Luong, M.-T., Kocisky, T., Goyal, N., Joshi, V., Chen, D., Levy, O., Zettlemoyer, L., & Stoyanov, V. (2020). *Multilingual Denoising Pre-training for Neural Machine Translation*. Proceedings of ACL.

Lin, C.-Y. (2004). *ROUGE: A Package for Automatic Evaluation of Summaries*. In Text Summarization Branches Out: Proceedings of ACL Workshop.