# SumeCzech: Large Czech News-Based Summarization Dataset

**Milan Straka, Nikita Mediankin, Tom Kocmi, Zdeněk Žabokrtský, Vojtěch Hudeček, Jan Hajič**
Institute of Formal and Applied Linguistics
Charles University, Faculty of Mathematics and Physics
Malostranské náměstí 25, Prague, Czech Republic
{straka,mediankin,kocmi,zabokrtsky,hudecek,hajic}@ufal.mff.cuni.cz

**Abstract**

Document summarization is a well-studied NLP task. With the emergence of artificial neural network models, the summarization performance is increasing, as are the requirements on training data. However, only a few datasets are available for Czech, none of them particularly large. Additionally, summarization has been evaluated predominantly on English, with the commonly used ROUGE metric being English-specific. In this paper, we try to address both issues. We present SumeCzech, a Czech news-based summarization dataset. It contains more than a million documents, each consisting of a headline, a several sentences long abstract and a full text. The dataset can be downloaded using the provided scripts available at http://hdl.handle.net/11234/1-2615. We evaluate several summarization baselines on the dataset, including a strong abstractive approach based on Transformer neural network architecture. The evaluation is performed using a language-agnostic variant of ROUGE.

**Keywords:** SumeCzech, summarization dataset, document summarization, ROUGE, Czech

## 1. Introduction

Similarly to many other NLP tasks, performance of automatic document summarization has been improving with the recent rise of neural network methods. While deep neural network models can leverage large datasets, only a few moderately-sized datasets are available for document summarization when compared to, e.g., machine translation.

Additionally, document summarization has been explored mostly on English, with the dominant ROUGE metric (Lin, 2004) being English-specific (utilizing English stemmer, stop words and synonyms).

In order to provide more data for document summarization in Czech, this paper introduces SumeCzech – a collection of one million Czech news articles, each consisting of a headline, a several sentence abstract and a full text. The documents originate from five Czech Internet news sites. The dataset can be downloaded using our provided scripts.

Headline-abstract-text structure of the documents allows the dataset to be used for multiple summarization setups: headline generation either from an abstract or a full text, or generation of a multi-sentence abstract from a full text.

To enable automatic evaluation of summarization for Czech, we also propose a straightforward language-agnostic variant of the ROUGE metric, which we call $\text{ROUGE}_{\text{RAW}}$.

We evaluate several baselines for all selected summarization settings. Apart from several unsupervised methods, we evaluate two supervised methods: an extractive one inspired by approach by Kupiec et al. (1995), and an abstractive baseline based on Transformers neural network architecture (Vaswani et al., 2017).

## 2. Related Work

### 2.1. Datasets

Sentence summarization has been traditionally connected with the task of headline generation. The task was standardized around the DUC-2003 and DUC-2004 competitions (Over et al., 2007), which provided a standard evaluation set consisting of 500 news articles from New York Times and Associated Press Wire, each paired with 4 different human-generated reference summaries. For training, the Gigaword dataset (Graff et al., 2003) has been used frequently, offering 4 million news articles including their headlines.

Recently, Nallapati et al. (2016a) modified the CNN/Daily Mail corpus constructed by Hermann et al. (2015) to serve for multi-sentence summarization. The corpus consists of approximately 300 000 documents. Additionally, Filippova and Altun (2013) proposed a method for constructing datasets for extractive sentence summarization.[1]

To our best knowledge, only small summarization datasets exist for Czech: Czech part of the MultiLing dataset (Giannakopoulos et al., 2015; Li et al., 2013; Elhadad et al., 2013) containing 40 Wikipedia articles, and SummEC (Rott and Červa, 2013) containing 50 news articles.

### 2.2. Metrics

ROUGE (Lin, 2004) is the most commonly used metric, proposed as an English-specific recall-based metric. It utilizes English stemmer, stop words and synonyms.

Recently, the METEOR metric (Denkowski and Lavie, 2014) has been used by See et al. (2017) to evaluate multi-sentence summarization.

### 2.3. Summarization Methods

Summarization methods are generally either *extractive* or *abstractive*. Extractive methods only select suitable parts (sentences, words or phrases) from the document, while abstractive methods can produce an arbitrary text as the summary.

The extractive summarization methods are typically unsupervised, for example Luhn (Luhn, 1958), Latent Se-

[1] The dataset has been recently released at https://github.com/google-research-datasets/sentence-compression.

mantic Analysis (Steinberger and Ježek, 2004), LexRank (Erkan and Radev, 2004), TextRank (Mihalcea and Tarau, 2004), SumBasic (Vanderwende et al., 2007) or KL-Sum (Haghighi and Vanderwende, 2009). However, very good results in extractive summarization were achieved recently with recurrent neural networks (Filippova et al., 2015; Filippova and Alfonseca, 2015; Nallapati et al., 2016b; Nallapati et al., 2017).

Abstractive approach relies predominantly on the machine translation paradigm, also boosted by the recent success of neural machine translation (Rush et al., 2015; Nallapati et al., 2016a; Gülçehre et al., 2016; See et al., 2017).

# 3. The Dataset

## 3.1. Choice of Data Sources

When designing the dataset, we considered two main requirements. First, and most importantly, we wanted to produce a dataset that would be sufficiently large for deep learning methods to be applicable to it. However, we possessed limited human and time resources making it impossible to accomplish this task by creating summaries manually. This implied an automatic or a semi-automatic method of collecting the data, facilitating the need for a data source consisting of documents that would already have some kind of easily identifiable human-produced summary. Second, we wanted the data to be more or less domain-neutral, i.e., without much domain-specific terminology.

Collecting a dataset of scientific articles using their abstracts as summaries was considered, but promptly rejected. The next choice was to use electronic newspapers as they seemed to be able to provide a reasonable amount of data with reasonably well separated short abstracts preceding the articles.

The raw data for the dataset was collected from the Common Crawl project[2] using the Common Crawl API. Initially, five Czech news websites were selected to create the dataset: `novinky.cz`, `lidovky.cz`, `denik.cz`, `idnes.cz`, and `ihned.cz`. However, during the cleanup of the data, we decided to drop `ihned.cz` from the dataset, because too many of its pages turned out to be just abridged versions of the actual articles with links to paid content. Instead, `ceskenoviny.cz`, which provides mostly high-quality articles, was added to the collection.

## 3.2. Data Preparation

The data was prepared in the following steps:

1. Dumps of the relevant websites' pages from 10 Common Crawl collections were downloaded.
2. Irrelevant entries such as advertisement pages, article listings and photo galleries were filtered out based on a set of simple heuristics.
3. From each seemingly relevant entry, its headline, abstract and full text were extracted based on the HTML structure of the webpage, cleaned from HTML markup, embedded javascript and irrelevant information such as:
   - advertisement links;
   - links to other news;
   - leftover captions of embedded photo and video materials;
   - low-level headers embedded in the text, which are used as paragraph titles in some texts but should be removed because they are not really part of the text.
4. Frequently seen leading tags such as FOTO, VIDEO, country, city were removed from headlines and abstracts. These tags were usually separated from the rest of the headline or abstract by a dash or a colon (e.g., "Praha: ..."). For the purpose of cleaning these up, lists of most frequent tokens seen at the start of headlines and abstracts before dash or colon were created and manually checked. Names of persons with the following colon (indicating direct speech) were deliberately left in place.
5. The following documents were dropped:
   - with empty headline;
   - with abstract shorter than 10 words;
   - with full text shorter than 100 words;
   - with text-to-abstract ratio less than 4.
6. Language recognition was performed with langdetect,[3] Python port of Google's language-detection library,[4] and non-Czech documents were dropped.
7. A number of documents was dropped based on the headline and/or abstract text (e.g., some headlines clearly indicated that the page is an advertisement, not a news article, some abstracts were disclaimers that the page belongs to a series of culinary recipes with no other information in the abstract).
8. A number of documents was dropped based on the presence of certain keywords in the headline or abstract, e.g., some abstracts were starting with the word 'aktualizováno' ('updated'), a metainformation not directly connected with the content of the article that could not be reliably removed.
9. From the sets of documents with either duplicate headlines, duplicate abstracts or duplicate texts, only one document was retained. Therefore, headlines in the dataset are unique, as well as abstracts and texts.
10. Some inexact news duplicates were filtered out based on several heuristics. Specifically for `denik.cz`, all regional pages were dropped based on their URLs, since they were mostly either reprints of central news or very specific entries such as "Where to play football this weekend".
11. Date of each article's publication was extracted wherever possible either from the page's metadata or from its body based on HTML markup. All dates were then converted into standardized format.

## 3.3. Structure of Dataset Entries

The dataset is produced in the JSON Lines format,[5] where each document is represented on a single line as a JSON object with the following fields:

[2] `http://commoncrawl.org`

[3] `https://pypi.python.org/pypi/langdetect`

[4] `https://github.com/shuyo/language-detection`

[5] `http://jsonlines.org`

| Website | Documents | |
|---|---|---|
| | Number | Percentage |
| ceskenoviny.cz | 4 854 | 0.5% |
| denik.cz | 157 581 | 15.7% |
| idnes.cz | 463 192 | 46.2% |
| lidovky.cz | 136 899 | 13.7% |
| novinky.cz | 239 067 | 23.9% |
| Total | 1 001 593 | |

Table 1: Number of documents from individual websites.

- `url`: the URL of the article webpage from where it was crawled by the Common Crawl;
- `headline`;
- `abstract`;
- `text`;
- `subdomain`: some of the source websites have clear-cut subdomains for different broad topics, e.g., `lidovky.cz` has `sport.lidovky.cz` for sport news; these were extracted from article URLs for possible future use as a surrogate means of identificating a human-assigned article topic;
- `section`: another option for topic identification: sometimes, a broad topic can be extracted from the part of the URL that follows the domain name part;
- `published`: date of publication in RFC 3339 format,[6] with all dates stored in CET and CEST as appropriate (i.e., utilizing the timezone in which the article was published).

Headlines and abstracts are stored without any line breaks. The former mostly did not have them originally, while the latter either had none or had each sentence separated by a line break, depending on the website formatting, making line breaks in abstracts non-indicative. Line breaks in full text are used as the means to separate the paragraphs of the original text.

We put the emphasis on maximum human-readability of the resulting data without sacrificing the ease of processing. Both the former and the latter were the reasons behind choosing JSON Lines format. First, Czech uses significant amount of accented characters, therefore it was important to be able to save the data in UTF-8 character encoding as is, i.e., without escaping non-ASCII characters, which is permitted in JSON Lines. Second, we wanted to keep individual entries contained within single lines to facilitate the ease of use of the dataset with Unix-style text processing tools.

### 3.4. Dataset Size Statistics

In total, the dataset contains approximately one million documents, with the distribution across websites shown in Table 1.

The quantitative statistics of headline, abstract and full text length are displayed in Table 2. The headlines are approximately 9 words long on average, with the abstracts being nearly five times the size and the full texts being nearly ten times the size of abstracts.

[6] `https://www.ietf.org/rfc/rfc3339.txt`

| | Q1 | Median | Q3 | Mean | Stddev |
|---|---|---|---|---|---|
| Headlines | 7 | 9 | 11 | 9.4 | 2.9 |
| Abstracts | 33 | 42 | 51 | 42.2 | 14.8 |
| Texts | 265 | 378 | 553 | 470.1 | 365.3 |

Table 2: Quantitative statistics of lengths of headlines, abstracts and texts in words. Q1 and Q3 denote the first and the third quartile, respectively.

### 3.5. Dataset Split

Before splitting the data into train, dev and test sets, we theorized that having too similar documents in the train and the test sets could lead to a skewed (too optimistic) evaluation of any supervised summarization methods. Therefore, we wanted the documents that are close to each other in some sense to be put into the same part of the split. At the same time, we did not want to end up with all the documents from one domain in the same part of the split, as it would introduce even stronger bias to the evaluation. To elaborate, this can be imagined as a situation when a model is trained on the data from one domain and then evaluated on the data from another. However, it appeared to us that the possibility of evaluation on an out-of-domain test set would be an interesting option. This, again, can be thought of as a common real-life situation when a model is trained on the data from one domain, then used on real data from other domain. In this use case having an out-of-domain test set could provide some insight into the model's possible behavior on real-world data.

Taking into account the above considerations, we devised the following procedure. The documents were first clustered into 25 clusters by K-Means algorithm, based on normalized L2 similarity of their abstracts. A cluster of size approximately 4.5% of the whole dataset size was taken as the out-of-domain test set. The rest of the data was then clustered again into 5000 clusters by K-Means algorithm, again based on L2 similarity of their abstracts. Consequently, the clusters were randomly divided in rougly 86.5:4.5:4.5 ratio to form the standard train/dev/test split.

The sizes of the individual dataset parts, along with distribution of articles across websites in each part, are presented in Table 3.

When inspected, the out-of-domain test set turned out to contain news about concerts and festivals, which is indeed out of domain when related to other topics, albeit not radically, because it is still news articles.

## 4. Obtaining the SumeCzech Dataset

Instead of distributing the produced dataset, we provide the two components for an end user to recreate it: the document listings and the extractor script.

The document listings contain the following values for each documents of the dataset:

- name of the Common Crawl file that contains the raw data for the document;
- its offset in the Common Crawl file;
- its length in the Common Crawl file;

| Website | Documents | |
|---|---|---|
| | Number | Percentage |
| train | | |
| ceskenoviny.cz | 4318 | 0.5% |
| denik.cz | 137926 | 15.9% |
| idnes.cz | 404367 | 46.6% |
| lidovky.cz | 118761 | 13.7% |
| novinky.cz | 202224 | 23.3% |
| Total | 867596 | |
| dev | | |
| ceskenoviny.cz | 229 | 0.5% |
| denik.cz | 7559 | 17.0% |
| idnes.cz | 21163 | 47.5% |
| lidovky.cz | 5755 | 12.9% |
| novinky.cz | 9861 | 22.1% |
| Total | 44567 | |
| test | | |
| ceskenoviny.cz | 168 | 0.4% |
| denik.cz | 6854 | 15.4% |
| idnes.cz | 19960 | 44.9% |
| lidovky.cz | 6462 | 14.5% |
| novinky.cz | 11010 | 24.8% |
| Total | 44454 | |
| out-of-domain test | | |
| ceskenoviny.cz | 139 | 0.3% |
| denik.cz | 5242 | 11.7% |
| idnes.cz | 17702 | 39.4% |
| lidovky.cz | 5921 | 13.2% |
| novinky.cz | 15972 | 35.5% |
| Total | 44976 | |

Table 3: The train/dev/test/out-of-domain test split of SumeCzech.

- which set (train/dev/test/out-of-domain test) this document belongs to;
- MD5 sum of the corresponding entry in the dataset.

The first three values deterministically define the place of the raw data for the document in the Common Crawl data and allow for its retrieval via Common Crawl API. The last value allows to check if the extraction procedure have successfully recreated the document from the raw data.

The extractor script is written in Python 3 and recreates the dataset using the document listings and the Common Crawl data by downloading the raw data and applying the original steps described in 3.2. that are required to extract headlines, abstracts, full texts and metadata and clean them up (but not the steps involved in filtering out undesirable documents, because those documents are already absent from the listings). The script then checks each recreated entry against the corresponding MD5 sum provided in the listings.

The document listings and the extraction script are available for download at `http://hdl.handle.net/11234/1-2615` under Mozilla Public License 2.0.[7]

We do not impose any additional licensing restrictions on the recreated dataset, however, it is subject to the Common Crawl terms of use,[8] and, by extension, local legislations regulating authors' rights that are in effect in the end user's country.

## 5. Evaluation Metrics

A standard way to evaluate summarization task is to use the ROUGE metric (Lin, 2004). ROUGE is an English-specific metric (employing English stemmer, stop words and synonyms), and was originally recall-based. In the DUC task, both the gold summary and the system summary is capped at 75 bytes and the recall of the non-stop words is evaluated, taking synonyms into account.

However, with the appearance of other datasets and more powerful abstractive methods, a fixed limit on the summary length became neither desirable nor needed, and, therefore, full-length F1 ROUGE is also being used recently (Nallapati et al., 2016a; Chopra et al., 2016; See et al., 2017).

Therefore, we propose to evaluate summarization methods trained on the SumeCzech dataset using full-length F1-score of a language-agnostic variant of ROUGE, which utilizes no stemmer, no stop words and no synonyms. We denote this variant $\text{ROUGE}_{\text{RAW}}$ and report $\text{ROUGE}_{\text{RAW}}$-1 (unigrams), $\text{ROUGE}_{\text{RAW}}$-2 (bigrams) and $\text{ROUGE}_{\text{RAW}}$-L (longest common subsequence). The Python 3 implementation of language-agnostic $\text{ROUGE}_{\text{RAW}}$ is provided alongside the SumeCzech dataset.

## 6. Experiments

The dataset allows for three summarization task setups:

- abstract→headline: generate one-sentence summary using a paragraph of approximately 3 sentences; similar to the DUC (Over et al., 2007) and Gigaword (Graff et al., 2003) tasks;
- full text→headline: generate one-sentence summary using a full text of several dozen sentences; also similar to the DUC (Over et al., 2007) and Gigaword (Graff et al., 2003) tasks;
- full text→abstract: generate multi-sentence summary using a full text consisting of several dozen sentences; similar to the CNN/Daily Mail (Nallapati et al., 2016a) task.

### 6.1. Extractive Methods

#### 6.1.1. Unsupervised

We evaluate several unsupervised extractive methods for all three summarization setups. All methods extract either 1 or 3 sentences, depending on whether they are generating a headline or an abstract, respectively. We employed the following methods:

- `first`: return given number (1 or 3) of initial sentences. Such baseline, while seemingly trivial, usually achieves high performance on news articles and is very hard to beat, because authors tend to summarize the most prominent information in the first few sentences.
- `random`: return randomly chosen sentences.
- `textrank`: TextRank (Mihalcea and Tarau, 2004), a classic unsupervised method based on the representation of the text as a network of sentences based on their similarity.

[7] `http://www.mozilla.org/MPL/2.0/`

[8] `http://commoncrawl.org/terms-of-use/full`

For the above methods, we use our own Python 3 implementation. TextRank utilizes a list of Czech stop words for the purposes of calculating sentence similarity.

#### 6.1.2. Supervised

In order to evaluate supervised approach, we include an extractive machine learning method inspired by the work of Kupiec et al. (1995). In this method, we first transform each sentence to a vector of features that are listed below:

- `TF-IDF` (Ramos and others, 2003): sum of `TF-IDF` measured for each word normalized by the sentence length. In the inference phase, we rely on the frequency values obtained during training.
- `Length`: length of the sentence.
- `Cohesion`: total distance from the sentence to the other ones in terms of edit distance.
- `Proper names`: count of capitalized words in the sentence.
- `Numbers`: count of tokens that consist of digits.
- `Non-essential words`: count of common words that indicates that the sentence relates to some other one.

In the training phase, the vectors are labeled by binary values. First, the sentences are sorted based on their similarity to the sentences from the gold abstract (or headline, respectively). Then, top sentences are picked and corresponding feature vectors are marked positive, the rest is considered negative. This way we obtain a classification task and we can train a classifier. We consider two classification algorithms: logistic regression and random forests. In the inference phase, the sentences are transformed into vectors again, and the classifier assigns each one the probability of being picked. Finally, a fixed number of sentences with the best scores is picked.

Depending on the employed classifier, the method is dubbed either `clf-lr` (when classifier used is logistic regression) or `clf-rf` (when random forests are employed).

### 6.2. Abstractive Summarization

Following the recent success in abstractive summarization (See et al., 2017), we also evaluated an abstractive summarization method. We utilized the tensor2tensor framework,[9] namely version 1.2.9. We used a neural machine translation model of Vaswani et al. (2017) with hyperparameters set as in model called `base` in the paper.[10] We evaluated the abstractive summarization method, dubbed `t2t`, on all three tasks.

We trained the model on the lowercased data and vocabulary of 32 000 word-pieces (Wu et al., 2016). We utilized GeForce GTX 1080 Ti GPU for training. The batch sizes differed for each task, batch size of 1700 was used for abstract→headline, batch size of 6500 for text→abstract and batch size of 7500 for text→headline. The final models utilize averaging over last 8 consecutive checkpoints (one hour from each other). For the abstract→headline task, we trained the model for 15 days and for the final evaluation we use beam size 4. The tasks text→headline/abstract were trained for 8 days, use beam size 3 and clip all inputs to maximal length of 400 words in order to fit in GPU memory.

[9] `https://github.com/tensorflow/tensor2tensor`

[10] The `big` model as described in the paper exhibited worse results, possibly due to a small maximal batch size.

| Method | $ROUGE_{RAW}$-1 | | | $ROUGE_{RAW}$-2 | | | $ROUGE_{RAW}$-L | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F | P | R | F | P | R | F |
| test | | | | | | | | | |
| `first` | 13.3 | 22.9 | 15.9 | 4.3 | 7.6 | **5.2** | 11.9 | 20.5 | 14.3 |
| `random` | 10.0 | 16.6 | 11.6 | 2.7 | 4.7 | 3.2 | 9.0 | 14.8 | 10.4 |
| `textrank` | 12.9 | 22.4 | 15.5 | 4.1 | 7.3 | 4.9 | 11.6 | 20.0 | 13.9 |
| `clf-lr` | 11.5 | **29.6** | 15.9 | 3.4 | **9.3** | 4.7 | 9.8 | **25.4** | 13.7 |
| `t2t` | **19.3** | 15.4 | **16.6** | **6.2** | 4.8 | **5.2** | **17.9** | 14.3 | **15.4** |
| out-of-domain test | | | | | | | | | |
| `first` | 13.5 | 25.1 | **16.6** | 4.8 | 9.3 | **5.9** | 12.1 | 22.4 | 14.8 |
| `random` | 10.2 | 18.7 | 12.4 | 3.1 | 6.2 | 3.9 | 9.2 | 16.7 | 11.1 |
| `textrank` | 13.2 | 24.7 | 16.2 | 4.6 | 9.0 | 5.7 | 11.8 | 21.9 | 14.5 |
| `clf-lr` | 11.5 | **28.6** | 15.3 | 3.9 | **10.7** | 5.4 | 9.9 | **24.5** | 13.1 |
| `t2t` | **18.9** | 14.8 | 16.0 | **6.8** | 5.0 | 5.5 | **17.7** | 13.9 | **15.0** |

Table 4: Abstract→headline summarization results.

### 6.3. Results and Discussion

We evaluated the above extractive and abstractive methods on both the test and out-of-domain test portions of SumeCzech, utilizing the $ROUGE_{RAW}$-1, $ROUGE_{RAW}$-2 and $ROUGE_{RAW}$-L metrics. To allow for more detailed interpretation of the results, we present not only F1-score, but also precision and recall.

Before we present the results, it is worth mentioning that the `first` baseline is usually very difficult to overcome, especially in the domain of news articles (Nallapati et al., 2016a; See et al., 2017).

First, we present the evaluation of extractive and abstractive methods in the abstract→headline setting in Table 4. The extractive methods perform similarly to `first` baseline, but the `first` baseline has slightly higher F-scores. The abstractive `t2t` method performs the best, achieving the highest F-scores in all three $ROUGE_{RAW}$ variants.

Note that the abstractive method has very high precision, but lacks in recall. We found out that this is a consequence of generating too short headlines. While the gold headlines have an average length of 9.7 words, the headlines generated by the `t2t` method consist of 7.7 words on average. We therefore conclude that a higher performance could be achieved by better matching the length distribution of the headlines.

On the out-of-domain test set, the results of the `t2t` method are lower relative to the performance of other algorithms. Notably, the F-score of the `first` baseline is the highest for $ROUGE_{RAW}$-1 and $ROUGE_{RAW}$-2 metrics, while being only slightly behind the best $ROUGE_{RAW}$-L F-score, which was achieved by `t2t`. We hypothesise that this drop is caused by the `t2t` method not being able to generalize well enough for the out-of-domain test set.

The results of summarization of full texts into headlines are presented in Table 5. Both supervised algorithms `clf-rl` and `t2t` demonstrate lower F-score performance than the unsupervised `first` and `textrank` methods. However, the precision of `t2t` approach still surpasses all other meth-

| Method | ROUGE$_{RAW}$-1 | | | ROUGE$_{RAW}$-2 | | | ROUGE$_{RAW}$-L | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F | P | R | F | P | R | F |
| test | | | | | | | | | |
| first | 6.3 | 11.8 | **7.6** | **1.1** | 2.2 | **1.4** | 5.7 | 10.6 | **6.8** |
| random | 4.3 | 8.0 | 5.2 | 0.5 | 1.0 | 0.6 | 4.0 | 7.2 | 4.7 |
| textrank | 5.6 | **15.3** | **7.6** | 0.9 | **2.6** | 1.2 | 4.9 | **13.3** | 6.6 |
| clf-rf | 5.0 | 9.4 | 6.3 | 0.7 | 1.3 | 0.8 | 4.5 | 8.4 | 5.6 |
| t2t | **7.4** | 5.9 | 6.4 | 0.7 | 0.5 | 0.6 | **7.0** | 5.6 | 6.0 |
| out-of-domain test | | | | | | | | | |
| first | **6.2** | 12.2 | 7.6 | **1.3** | 2.6 | **1.6** | **5.6** | 10.9 | 6.8 |
| random | 4.3 | 8.3 | 5.2 | 0.6 | 1.2 | 0.7 | 3.9 | 7.5 | 4.7 |
| textrank | 5.7 | **15.9** | **7.8** | 1.1 | **3.3** | 1.5 | 5.0 | **13.9** | **6.9** |
| clf-rf | 5.3 | 10.1 | 6.7 | 1.0 | 2.1 | 1.3 | 4.9 | 9.3 | 6.1 |
| t2t | 5.3 | 4.3 | 4.6 | 0.4 | 0.3 | 0.4 | 5.0 | 4.1 | 4.3 |

Table 5: Text→headline summarization results.

| Method | ROUGE$_{RAW}$-1 | | | ROUGE$_{RAW}$-2 | | | ROUGE$_{RAW}$-L | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F | P | R | F | P | R | F |
| test | | | | | | | | | |
| first | **13.3** | 18.4 | **14.6** | **2.3** | 3.4 | **2.6** | 8.9 | 12.4 | **9.8** |
| random | 11.6 | 15.5 | 12.5 | 1.4 | 2.1 | 1.6 | 7.7 | 10.4 | 8.3 |
| textrank | 11.6 | 21.5 | 14.3 | 1.9 | 3.8 | 2.4 | 7.6 | 14.1 | 9.3 |
| clf-rf | 10.5 | **23.3** | 13.8 | 1.6 | **3.9** | 2.2 | 6.7 | **15.0** | 8.8 |
| t2t | 12.2 | 9.4 | 10.2 | 1.1 | 0.8 | 0.9 | **9.6** | 7.4 | 8.0 |
| out-of-domain test | | | | | | | | | |
| first | **12.2** | 18.1 | **13.8** | **2.1** | 3.4 | **2.5** | 8.3 | 12.4 | **9.3** |
| random | 10.7 | 15.4 | 11.9 | 1.4 | 2.2 | 1.6 | 7.3 | 10.5 | 8.1 |
| textrank | 11.0 | **21.2** | 13.7 | 2.0 | **4.0** | **2.5** | 7.3 | **14.2** | 9.1 |
| clf-rf | 9.1 | 20.2 | 11.9 | 1.4 | 3.3 | 1.8 | 6.3 | 13.5 | 7.9 |
| t2t | 11.7 | 8.4 | 9.3 | 0.8 | 0.6 | 0.7 | **9.6** | 7.0 | 7.7 |

Table 6: Text→abstract summarization results.

ods in two ROUGE variants.

Similarly to the previous settings, the performance of t2t deteriorates on the out-of-domain test set, while other methods are mostly unaffected.

The last considered setup of text→abstract summarization is evaluated in Table 6, yielding results similar to the previous setup. The first baseline is performing the best, followed by the textrank approach. The relative performance of the t2t abstractive summarization is the lowest, being inferior even to the random baseline on both test and out-of-domain test sets.

In order to compare quality of documents from different websites, we also analyse the first baseline in the abstract→headline setup for every website separately. The results are presented in Table 7. The ROUGE$_{RAW}$ metric shows that all websites provide headlines of similar quality, with the exception of ceskenoviny.cz, which provides headlines that are much more similar to the first sentences of their articles' abstracts.

### 6.4. Examples

We illustrate three test set examples of first and t2t baselines in abstract→headline setup in Figure 1. In order to make the examples accessible to non-Czech speaking audience, we translated the examples to English, preserving the original phrase structure and vocabulary as much as possible.

In all examples, the first method produces a good summary, even though quite large. The t2f method generates fluent summaries of suitable length, but while in the first case the headline is identical to the gold one, in the second case it is slightly paraphrased, and in the third case the produced headline uses completely different words than the gold one. Even while the headline produced by the t2t method is of high quality in all three cases, it receives lower ROUGE$_{RAW}$ score in the second case and zero score in the third case.

| Website | ROUGE$_{RAW}$-1 F-score | ROUGE$_{RAW}$-2 F-score | ROUGE$_{RAW}$-L F-score |
|---|---|---|---|
| test | | | |
| ceskenoviny.cz | 29.8 | 13.8 | 27.9 |
| denik.cz | 16.6 | 6.1 | 15.1 |
| idnes.cz | 14.2 | 4.2 | 12.6 |
| lidovky.cz | 16.5 | 5.3 | 14.8 |
| novinky.cz | 18.2 | 6.1 | 16.2 |
| All websites | 15.9 | 5.2 | 14.3 |
| out-of-domain test | | | |
| ceskenoviny.cz | 30.7 | 14.4 | 27.6 |
| denik.cz | 16.2 | 6.1 | 14.6 |
| idnes.cz | 14.8 | 4.8 | 13.1 |
| lidovky.cz | 17.8 | 6.3 | 15.7 |
| novinky.cz | 18.7 | 7.2 | 16.8 |
| All websites | 16.6 | 5.9 | 14.8 |

Table 7: The first baseline for abstract→headline task computed per website.

## 7. Conclusions

We have presented SumeCzech, a new large news summarization dataset for Czech. Every document in the dataset is composed of a short headline, an abstract comprising a few sentences, and a full text, allowing for several summarization setups. The scripts for downloading the dataset are available at http://hdl.handle.net/11234/1-2615.

We use language-agnostic variant of ROUGE metric ROUGE$_{RAW}$ for evaluation.

Finally, we have evaluated several baseline extractive summarization methods, both unsupervised and supervised, as well as an abstractive method based on neural machine translation Transformer architecture with subword units (Vaswani et al., 2017).

## Acknowledgments

This work has been partially supported by grants GAUK 1114217/2017, GAUK 8502/2016, and SVV 260 453 grant of Charles University.

This work has also been partially supported by, and data has been stored into, the LINDAT/CLARIN repository, a large research infrastructure supported by the Ministry of Education, Youth and Sports of the Czech Republic under projects LM2015071 and CZ.02.1.01/0.0/0.0/16_013/0001781.

| Method | Headline |
|---|---|
| gold | Žalobce navrhl pro Sisáka a Halu vazbu<br>*The prosecutor proposed remand for Sisak and Hal* |
| first | Státní zástupce Adam Borgula navrhl poslat finančníka Petra Sisáka a jeho pravou ruku, advokáta Iva Halu, do vazby.<br>*State attorney Adam Borgula proposed to send financier Peter Sisak and his right hand, lawyer Iva Hal, to remand.* |
| t2t | Žalobce navrhl pro sisáka a halu vazbu<br>*The prosecutor proposed remand for sisak and hal* |
| gold | Sněmovna dala šanci úplnému zákazu kouření v restauracích<br>*The parliament gave a chance to a complete smoking ban in restaurants* |
| first | Sněmovna dala šanci v dnešním úvodním kole úplnému zákazu kouření cigaret v restauracích, barech, vinárnách nebo v kavárnách a čajovnách.<br>*In today's opening round, the parliament gave a chance to a complete smoking ban of cigarettes in restaurants, bars, wine bars, cafes and tearooms.* |
| t2t | Poslanci dali šanci zákazu kouření v restauracích<br>*The deputies gave a chance to a smoking ban in restaurants* |
| gold | Rumunsko přijme prvky amerického raketového štítu<br>*Romania will accept elements of American rocket shield* |
| first | Rumunská nejvyšší rada obrany (CSAT) ve čtvrtek schválila plán Spojených států rozmístit v Rumunsku pozemní prvky nového systému protiraketové obrany.<br>*On Thursday, the Romanian Supreme Defense Council (CSAT) approved the United States' plan to distribute in Romania the ground elements of the new anti-missile defense system.* |
| t2t | Rumuni schválili nový protiraketový systém<br>*The Romanians approved a new anti-missile system* |

Figure 1: Examples of `first` and `t2t` methods in the abstract→headline setup taken from the test set. The English translations are in italics.

## 8. Bibliographical References


Chopra, S., Auli, M., and Rush, A. M. (2016). Abstractive sentence summarization with attentive recurrent neural networks. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 93–98. The Association for Computational Linguistics.

Denkowski, M. and Lavie, A. (2014). Meteor universal: Language specific translation evaluation for any target language. In *EACL 2014 Workshop on Statistical Machine Translation*.

Elhadad, M., Miranda-Jiménez, S., Steinberger, J., and Giannakopoulos, G. (2013). Multi-document multilingual summarization corpus preparation, part 2: Czech, hebrew and spanish. *MultiLing 2013*, page 13.

Erkan, G. and Radev, D. R. (2004). Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479.

Filippova, K. and Alfonseca, E. (2015). Fast k-best sentence compression. *CoRR*, abs/1510.08418.

Filippova, K. and Altun, Y. (2013). Overcoming the lack of parallel data in sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1481–1491. The Association for Computational Linguistics.

Filippova, K., Alfonseca, E., Colmenares, C. A., Kaiser, L., and Vinyals, O. (2015). Sentence Compression by Deletion with LSTMs. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 360–368. The Association for Computational Linguistics.

Giannakopoulos, G., Kubina, J., Conroy, J. M., Steinberger, J., Favre, B., Kabadjov, M. A., Kruschwitz, U., and Poesio, M. (2015). Multiling 2015: Multilingual summarization of single and multi-documents, on-line fora, and call-center conversations. In *SIGDIAL Conference*, pages 270–274.

Graff, D., Kong, J., Chen, K., and Maeda, K. (2003). English Gigaword. Philadelphia. Linguistic Data Consortium.

Gülçehre, Ç., Ahn, S., Nallapati, R., Zhou, B., and Bengio, Y. (2016). Pointing the unknown words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Haghighi, A. and Vanderwende, L. (2009). Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 362–370. Association for Computational Linguistics.

Hermann, K. M., Kociský, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., and Blunsom, P. (2015). Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1693–1701.

Kupiec, J., Pedersen, J., and Chen, F. (1995). A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 68–73. ACM.

Li, L., Forascu, C., El-Haj, M., and Giannakopoulos, G. (2013). Multi-document multilingual summarization corpus preparation, part 1: Arabic, English, Greek, Chinese, Romanian. Association for Computational Linguistics.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *In Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–

81.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2):159–165.

Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into text. In *EMNLP*, volume 4, pages 404–411.

Nallapati, R., Zhou, B., dos Santos, C. N., Gülçehre, Ç., and Xiang, B. (2016a). Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016*, pages 280–290. The Association for Computer Linguistics.

Nallapati, R., Zhou, B., and Ma, M. (2016b). Classify or select: Neural architectures for extractive document summarization. *CoRR*, abs/1611.04244.

Nallapati, R., Zhai, F., and Zhou, B. (2017). Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, pages 3075–3081. AAAI Press.

Over, P., Dang, H., and Harman, D. (2007). DUC in Context. *Inf. Process. Manage.*, 43(6):1506–1520, November.

Ramos, J. et al. (2003). Using TF-IDF to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 133–142.

Rott, M. and Červa, P. (2013). SummEC: A Summarization Engine for Czech. In *International Conference on Text, Speech and Dialogue*, pages 527–535. Springer.

Rush, A. M., Chopra, S., and Weston, J. (2015). A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 379–389. The Association for Computational Linguistics.

See, A., Liu, P. J., and Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1073–1083. Association for Computational Linguistics.

Steinberger, J. and Ježek, K. (2004). Using latent semantic analysis in text summarization and summary evaluation.

Vanderwende, L., Suzuki, H., Brockett, C., and Nenkova, A. (2007). Beyond sumbasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing & Management*, 43(6):1606–1618.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.