Universität Zürich[UZH]

Masterarbeit

zur Erlangung des akademischen Grades

Master of Arts

der Philosophischen Fakultät der Universität Zürich

Fine-tuning the SwissBERT Encoder Model for Embedding Sentences and Documents

Verfasser: Juri Grosjean

Matrikel-Nr: 15-562-382

Referent: Prof. Dr. Gerold Schneider

Betreuer: Dr. Jannis Vamvas

Institut für Computerlinguistik

Abgabedatum: 01.06.2024

Abstract

Encoder models trained for the embedding of sentences or short documents have proven useful for tasks such as semantic search and topic modeling. In this paper, a version of the SwissBERT encoder model specifically fine-tuned for this purpose is presented. SwissBERT contains language adapters for the four national languages of Switzerland – German, French, Italian, and Romansh – and has been pre-trained on a large number of news articles in those languages. Using contrastive learning based on a subset of the original training dataset, a fine-tuned version called SentenceSwissBERT was trained. Multilingual experiments on document retrieval, text classification, and topic modeling in a Switzerland-specific setting show that SentenceSwissBERT yields a better performance than the original model, as well as comparable baselines. The model is openly available for research use.[1]

Zusammenfassung

Für Textverarbeitungsaufgaben wie Semantic Search und Topic Modeling eignen sich Encoder-Modelle, welche für die Anwendung von Sentence Embeddings trainiert wurden. Die vorliegende Arbeit stellt eine derartige Version von SwissBERT vor. SwissBERT enthält modulare Adapter für die vier Schweizer Landessprachen – Deutsch, Französisch, Italienisch und Rätoromanisch. Das Modell wurde über eine grosse Anzahl von Nachrichtenartikeln in diesen Sprachen vortrainiert. Durch Contrastive Learning auf Basis eines Subsets des ursprünglichen Datensatz wurde eine Version von SwissBERT namens SentenceSwissBERT für Sentence Embeddings trainiert. SentenceSwissBERT zeigt eine bessere Performance als das originale Modell in diversen mehrsprachigen Evaluierungsaufgaben mit Schweizer Kontext, darunter Document Retrieval, Text Classification und Topic Modeling. Es schlägt ausserdem vergleichbare Baselines. Das Modell ist für Forschungszwecke frei zugänglich.[2]

[1] https://huggingface.co/jgrosjean-mathesis/sentence-swissbert

[2] https://huggingface.co/jgrosjean-mathesis/sentence-swissbert

Acknowledgement

I express my gratitude to Dr. Jannis Vamvas for his helpful supervision and valuable contribution to this paper. Furthermore, I thank Prof. Dr. Gerold Schneider for his insightful guidance. I also want to show my appreciation to Textshuttle for granting me access to their Romansh machine translation API for this research.

Contents

List of Figures

List of Tables

List of Acronyms

BERT	Bidirectional Encoder Representations from Transformers
CERT	Contrastive self-supervised Encoder Representations from Transformers
c-TF-IDF	class-based TF-IDF
DE	German
FR	French
HDBSCAN	Hierarchical Density-Based Spatial Clustering of Applications with Noise
IT	Italian
KNN	k-nearest-neighbor
LDAvis	Latent Dirichlet Allocation visualisation
LLM	large language model
LSTM	long-short-term memory
MLM	Masked Language Modeling
MultiNLI	Multi-Genre Natural Language Inference
NER	named entity recognition
NLP	natural language processing
NSP	Next Sentence Prediction
PMI	pointwise mutual information
RM	Romansh
RNN	recurrent neural network
RoBERTa	Robustly optimized BERT approach
RSI	Radiotelevisione svizzera di lingua italiana
RTR	Radiotelevisiun Svizra Rumantscha
RTS	Radio Télévision Suisse
SimCSE	simple contrastive sentence embedding framework
SNLI	Stanford Natural Language Inference
SRG SSR	Schweizerische Radio- und Fernsehgesellschaft / Société suisse de radiodiffusion et télévision / Società svizzera di radiotelevisione / Societad svizra da radio e televisiun

SRF	Schweizer Radio und Fernsehen
TF-IDF	term frequency-inverse document frequency
TM	topic modeling
UCI	University of California, Irvine
UMAP	Uniform Manifold Approximation and Projection
UMass	University of Massachusetts
X-MOD	Cross-lingual Modular

1 Introduction

Sentence embeddings have become a valuable tool in natural language processing. Neural models are fed with sequences of strings and convert them into embeddings, i.e. a numeric representation of the input text. These can be applied in a variety of contexts and tasks, e.g. information retrieval, semantic similarity, text classification and topic modeling. The general idea is that the embedding vectors can be applied for numeric calculations, which pure text data in a string format would not allow.

SwissBERT (Vamvas et al., 2023) is a modular encoder model based on X-MOD (Pfeiffer et al., 2022), which was specifically designed for multilingual representation learning. In this context, *modular* means that, while certain parts of the model are shared across all languages, there are certain components called *language adapters* that are specifically trained for one specific language only. SwissBERT has been trained via masked language modeling on more than 21 million Swiss news articles in Swiss Standard German, French, Italian, and Romansh Grischun. The model is designed for processing Switzerland-related text, e.g. for named entity recognition, part-of-speech tagging, text categorization, or word embeddings. Both X-MOD and SwissBERT are accessible via the Hugging Face Transformers library.[1][2]

The aim of this work is to fine-tune the existing SwissBERT model for the embedding of sentences and short documents, with the expectation that performance on these tasks will improve. Specifically, the hypothesis is that using the contrastive learning technique SimCSE (Gao et al., 2021) to fine-tune SwissBERT will yield a model that outperforms the base model as well as generic multilingual sentence encoders in the context of processing news articles from Switzerland.

This is evaluated on three natural language processing tasks that utilize sentence embeddings, namely document retrieval, text classification, and topic modeling. While all tasks are looked at from a monolingual perspective, the first two are additionally assessed from a cross-lingual perspective. The experiments show that the fine-tuned SwissBERT, which is called SentenceSwissBERT, has a higher accuracy than the

[1] https://huggingface.co/ZurichNLP/swissbert

[2] https://huggingface.co/facebook/xmod-base

baseline models. An especially strong effect was observed for the Romansh language, with an absolute improvement in accuracy of up to 55 percentage points over the original SwissBERT model, and up to 29 percentage points over the best Sentence-BERT baseline. The topic modeling outputs likewise show a strong performance of the newly fine-tuned model. The embeddings produced by SentenceSwissBERT generate topics that can be easily interpreted and outperform the baseline model in terms of perplexity and two topic coherence measures.

1.1 Thesis Structure

In chapter 1, an introduction and overview of the thesis was provided. Chapter 2 introduces research that is relevant to the given topic, namely explaining the theoretical background for encoder models, introducing various applications of embeddings, and according to which metrics the performance of a model carrying out these tasks can be measured. Chapter 3 elaborates on the exact training method that is applied for the fine-tuning of the SwissBERT model. It also introduces the exact evaluation tasks on which SentenceSwissBERT is assessed. In chapter 4, all results of the evaluation tasks are reported, comparing the performance of the new model to the baselines. Chapter 5 discusses the resulting figures and topics, analyses them and highlights the noteworthy. Finally, chapter 6 draws a conclusion on the underlying results and suggests ideas to further expand research on the topic.

2 Related work

This section introduces relevant literature that is directly related to the underlying research.

2.1 Embedding Spaces

The underlying concept of encoding into embedding spaces is vector semantics. A word or text sequence is represented as a point in a multi-dimensional semantic space (Jurafsky and Martin, 2024). These vector representations are called embeddings. Generally, text that serves a similar function or has a similar meaning should be represented by embeddings that are close to each other in the embedding space. Likewise, words or sequences that serve opposite purposes or are contradictory should be mapped far apart from each other.

Jurafsky and Martin (2024) mention that making use of embeddings brings an enormous amount of power to text processing tasks. It enables us to capture the meaning and context of a word or a text sequence. This is crucial when applying automatic text processing based on semantics, such as sentiment analysis. It also allows for semantic comparisons between two or multiple text strings, e.g. by calculating the dot product or cosine similarity of two embedding vectors.

In other words, utilizing embedding spaces in NLP is a way to apply mathematics, so that computers can grasp the meaning behind text data and, thus, are able to process it further in many different ways, just like humans.

2.2 Encoder Models

The way language is encoded and processed by computers has come a long way. Language or text has usually been regarded as a sequential, continuous, and temporal stream of data. Thus, it has generally been processed in a linear way (left-to-right

approach), e.g. via unidirectional recurrent neural networks (RNN) or long-short-term memory networks (LSTM). This changed since the introduction of bidirectional transformer encoders like BERT and its descendant RoBERTa, which always take the entire context of each token into consideration (left and right) and uses an additional dimension to take into account the position of the token in the input. This enables the machine learning models to understand the context of each token even better, which is crucial to process natural language in a meaningful way (Jurafsky and Martin, 2024).

The encoder-decoder architecture is a concept that many language models are based on. Phrased in a simplified way, it is a neural network that encodes text input into a meaningful numeric representation (called *context*) and, subsequently, produces a text output that is based on the original input by decoding this hidden state. This concept is especially popular for machine translation tasks (Jurafsky and Martin, 2024). However, the context does not necessarily have to be put through a decoder in order to make use of it.

The focus of bidirectional transformer-based encoders is to only produce contextualized representations of the input text tokens, i.e. they do not contain a decoder. They apply self-attention to turn strings of text into sequences of vectors that contain the input's meaning in a numeric format. These output embeddings store a great deal of semantic information that is tied to the specific context of the input tokens, including their relationship to all the other tokens in the input sequence. This means that the output embeddings are highly contextualized representations of each input token, which are highly useful for various NLP applications (Jurafsky and Martin, 2024).

2.2.1 BERT and RoBERTa

In order to create a neural network that encodes a given text input into meaningful output embeddings, a sensible architecture and training method has to be chosen. Devlin et al. (2019) present the Bidirectional Encoder Represenations from Transformers (BERT), which can serve as a base model to be fine-tuned for a vast variety of NLP tasks. The original BERT Base model entails the following:

- Training via Masked Language Modeling (MLM) and Next Sentence Prediction (NSP)
- English-only subword vocabulary consisting of 30 000 tokens
- Hidden layers with 768 dimensions respectively

- 12 layers of transformer blocks
- CLS token to represent entire input sequence

The Robustly optimized BERT approach (RoBERTa) by Liu et al. (2019) represents an expanded version of BERT, whose large variant entails the following:

- Training via Masked Language Modeling (MLM) only
- Multilingual subword vocabulary consisting of 250 000 tokens
- Hidden layers with 1 024 dimensions respectively
- 24 layers of transformer blocks
- CLS token to represent entire input sequence

Both BERT and RoBERTa use a fixed-size input of 512 subword tokens.

2.2.2 Sentence Embeddings

While embeddings for single words or tokens can be useful for certain NLP tasks, such as Named Entity Recognition (NER) or Part-of-Speech-Tagging, certain NLP taks, e.g. text classification or topic modeling, do better when utilizing a single, consolidated embedding for entire text sequences (Cer et al., 2018). The entire sequence should be used as an input, as it entails the complete meaning and context of the text, as opposed to single word / token embeddings. We can call this *sentence embedding*.

An example for the application of sentence embeddings is text classification (more on this in section 2.3.2). A simple approach to this involves training a single neural layer that maps the sentence embedding to the categories to which the text input should be classified (Logistic Regression). In recurrent neural networks, the hidden layer of the last input can be used as a base for this. In BERT and RoBERTa, the vector of the CLS token is originally suggested to be utilized for classification (Jurafsky and Martin, 2024).

However, there are also other strategies to derive sentence embeddings from the complete last hidden layer (all token embeddings) of an encoder output (pooling). Reimers and Gurevych (2019) compared three sentence embedding extraction approaches, namely using the output of the CLS token, computing the mean of all output vectors (MEAN pooling), and computing a max-over-time of the output token embeddings, which means taking the maximum value for each dimension across

all vectors (MAX pooling). They suggest using MEAN pooling per default, as it performs well across various different tasks.

2.2.3 Contrastive Learning

This technique was originally introduced in training neural models to perform vision tasks, e.g. image recognition. However, it has also been shown to deliver promising results with NLP tasks.[1] The goal is for the model to set up an embedding space in which similar data points are closely mapped to each other and dissimilar data points stay far apart from each other. For a mini-batch of N sentences, where (h_i, h_i^+) represent a pair of semantically-related sequences, h_j a random in-batch negative, and τ the temperature hyperparameter, the training objective looks as follows:

$$-\log \frac{e^{\text{cos\_sim}(h_i, h_i^+)/\tau}}{\sum_{j=1}^{N} e^{\text{cos\_sim}(h_i, h_j^+)/\tau}} \tag{2.1}$$

Contrastive learning was shown to be effective in the context of sentence embeddings that are useful for a variety of NLP tasks (Gao et al., 2021). Successful frameworks include Sentence-BERT (Reimers and Gurevych, 2019) (introduced in more detail in section 2.2.4 and CERT (Fang and Xie, 2020). Furthermore, contrastive learning enables data-efficient learning, e.g. in zero-shot settings (Zhang et al., 2022).

2.2.3.1 SimCSE

Introduced by Gao et al. (2021), the SimCSE (simple contrastive sentence embedding) framework has been found highly effective when used in conjunction with pre-trained language models. This technique can be applied using unsupervised or a supervised training.

For the unsupervised approach, the sequences in the training data are matched with themselves to create positive matches, i.e. the cosine similarity between both outputs (MEAN pooling or CLS) is maximized and, in parallel, the similarity to a random in-batch negative is minimized. Thanks to the dropout masks, the embeddings of identical sequences still differ slightly.

The supervised approach uses a dataset of sentence pairs with similar meanings and an optional third entry that is contradictory in meaning to the other two (hard

[1] https://paperswithcode.com/task/contrastive-learning

negative). The similarity computation is maximized for the similar sentence pairs and minimized between the positives and the negatives.

2.2.4 Sentence-BERT

Reimers and Gurevych (2019) suggest enhancing pre-trained models for generating fixed-size sentence embeddings. Their method applies siamese and triplet network architectures to finetune BERT or RoBERTa for this use. The training approach entails three objective functions: classification, regression, and triplet, each with specific training structures. This approach creates models that are able to set up high-quality sentence embeddings, e.g. for comparison via cosine similarity and more downstream tasks. Data from SNLI (Bowman et al., 2015) and MultiNLI datasets (Williams et al., 2018) was used for training. The results outperform all baselines on various NLP tasks. The Sentence-BERT method has given rise to a family of popular open-source encoder models.[2]

2.2.5 Multilingual Sentence Embeddings

There are multiple approaches for training encoder models to be able to process more than one language (multilingual text processing).

Yang et al. (2020) introduced the *Multilingual Universal Sentence Encoder for Semantic Retrieval* approach, which entails training the same model on 16 different languages in parallel via multiple tasks. It applies a dual-encoder framework, similar to Sentence-BERT (Reimers and Gurevych, 2019). The architecture is shared across all training languages. It was assessed according to various monolingual as well as cross-lingual evaluation tasks, with a particular focus on retrieval, and achieved promising results.

Reimers and Gurevych (2020) propose utilizing knowledge distillation to enhance monolingual models for multilingual use, which has proven highly effective. The approach entails using teacher model M in the source language s, a student model $\hat{M}$ to be trained for the target language t, and a set of parallel pairs $((s_1, t_1), \ldots, (s_n, t_n))$, in which t_i represents the translation of s_i in the target language. The teacher model sets up an embedding for the first sentence in the translation pair. This embedding is then used to train the student model: Both sequences of the translation pairs are embedded by the student model. The distance between each of the resulting em-

[2] https://www.sbert.net/

beddings respectively and the teacher model's original embedding is minimized via the mean-squared loss function to train the student model. For a given mini-batch $\mathcal{B}$, the formula presents as follows:

$$\frac{1}{|\mathcal{B}|}\sum_{j\in\mathcal{B}}\left[\left(M(s_j)-\hat{M}(s_j)\right)^2+\left(M(s_j)-\hat{M}(t_j)\right)^2\right] \tag{2.2}$$

Feng et al. (2022) have found that harnessing pre-trained language models and fine-tuning them for cross-lingual tasks yields high-quality outputs while requiring less training data than training encoder models from scratch via multilingual language data like translations.

2.2.5.1 X-MOD

Pfeiffer et al. (2022) propose the Cross-Lingual Modular model approach (X-MOD). They note that multilingual models oftentimes start performing worse the more languages are added to them. Their approach entails adding language-specific layers called *language adapters* to the encoder architecture during pre-training. These are each trained on one language respectively. This means that the language has to be specified when passing the input text into the model, so that it knows which modular component should be activated. The X-MOD approach creates a model that can easily be extended by adding new languages without the trade-off of this affecting its performance in the original languages, i.e. it is optimized for multilingual scalability.

2.2.5.2 SwissBERT

Vamvas et al. (2023) present SwissBERT, a masked language model created specifically for processing Switzerland-related text. It utilizes the X-MOD architecture to tackle the challenge of processing the four national languages of Switzerland (German, French, Italian, and Romansh). This means that the model includes four language-specific modular components and a particular sub-word vocabulary. It was trained using Swiss news articles only and intended to be used for Switzerland-related NLP tasks, especially showing promising results in named entity recognition (NER).

2.3 Evaluating Sentence Embedding Performance

Sentence embeddings can be used in a variety of different contexts and NLP tasks. As the goal is to capture the context-based semantic meaning of text sequences, typical downstream tasks to evaluate sentence embeddings are based on textual semantics and oftentimes apply similarity metrics like the cosine similarity or the dot product. Examples include classification, retrieval, semantic textual similarity, and topic modeling (Perone et al., 2018).

2.3.1 Document Retrieval

Retrieval describes a process that is similar (or sometimes identical) to a search function. During a document retrieval task, a text string query is put into the search system, which then returns a ranked set of documents from a collection. The goal is for the algorithm to output a document that matches the query the best. In other words, the system should accept a query, skims through a given corpus and returns the document that the user is most likely looking for (Jurafsky and Martin, 2024).

Jurafsky and Martin (2024) mention that retrieval is usually measured according to precision, recall, and F1-score.

Precision shows how well a given search system is able to find only documents that are relevant to the query, i.e. it measures the correctly found documents (true positives) against the ones it missed (false negatives):

$$\text{precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}} \tag{2.3}$$

In this context, the recall metric measures how well the search system can return all the relevant documents from the corpus. i.e. it measures the true positives against the documents it falsely deemed relevant (false positives):

$$\text{recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}} \tag{2.4}$$

The F1-score combines both recall and precision into one value:

$$\text{F1-score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{2.5}$$

A document retrieval system can make use of an embedding space. When doing this, both the queries and the documents are embedded. Subsequently, a similarity metric is applied to rank the potential documents based on how well they match with the query. In the context of sentence embeddings, this vector space used for a retrieval task is based on the sentence embeddings produced by the given encoder model (Jurafsky and Martin, 2024).

2.3.2 Text Classification

Text classification is a common NLP task in which a text is mapped to a finite set of given labels. Subgenres of this include sentiment analysis (mapping a text to a set of sentiments), language detection (mapping a text to a set of languages) and authorship detection (mapping a text to a set of authors) (Jurafsky and Martin, 2024).

The evaluation metrics for text classification are similar to the ones for retrieval, namely precision, recall and F1-score (Jurafsky and Martin, 2024). If the categorization includes more than two categories, these metrics are calculated for every single one of them and the average across all of them can be reported. A metric that encapsulates all of these aspects into a single number is the weighted-average F1-score, which is calculated according to this formula:

$$F1_{\text{weighted}} = \sum_{i=1}^{N} w_i \cdot F1_i \tag{2.6}$$

Here, N is the number of categories, w_i is the real number of occurences of the respective category i divided by total number of samples (a figure between 0 and 1), and $F1_i$ is the F1 score of category i.

In order to assess the model's capabilities internally, a train-test split is usually performed among the data. For example, 80% of the data is used to train a model (training data) and 20% is used to assess its performance (test data).

There are a great many different architectures that can be implemented for classifying texts. For instance, in the context of implementing categorization via machine learning, logistic regression and k-nearest-neighbor (KNN) algorithms can be applied.

2.3.2.1 Logistic Regression

This approach entails training a neural model / layer for classification, using a dataset that was manually categorized. The model is trained so that it outputs a probability for each of the given categories based on the input text. The category with the highest probability will be used as the predicted label. In the context of sentence embeddings, this system can be represented by only one neural layer that maps a given sentence embedding to the probabilities for each category. Logistic regression systems can be made use of for binary as well as multinomial classification (Jurafsky and Martin, 2024).

2.3.2.2 K-Nearest-Neighbor (KNN)

The straight-forward KNN method can also be used for text classification via an embedding space. A pre-labeled dataset of texts can be used for training, during which all documents are mapped to vectors and stored. When a new text that needs classification is inputted into the system, it is first mapped to an embedding. Then, the system searches for the k vectors from the training data that are closest to it in the embedding space, e.g. by applying cosine similarity. It can then be given the same label as its nearest neighbor in the embedding space (1-nearest-neighbor). The idea is that if a text is similar in meaning as one in the training data, it ought to belong to the same category. This is comparable to the retrieval method introduced in section 2.3.1.

2.3.3 Topic Modeling

Topic modeling (TM) is an NLP task that can also be regarded as a type of information retrieval. A topic modeling system takes a collection of documents as an input and then detects overarching topics in the dataset it was fed with. The topics should represent coherent clusters of semantically-related key words that are interpretable by humans (Abdelrazek et al., 2022).

According to Rüdiger et al. (2022), perplexity is the most suitable metric for the internal evaluation of topic models. The coherence of generated topics can be assessed according to the UCI and UMass metrics (Stevens et al., 2012).

2.3.3.1 Perplexity

Perplexity measures how well a model is able to generalize and predict new documents. A low score indicates that the model effectively creates topics and thus generalizes well (Ding et al., 2018). It is defined as the mean log-likelihood of words in a test corpus:

$$Perplexity(D_{test}) = \exp\left(-\frac{\sum_{d=1}^{M} \log p(w_d)}{\sum_{d=1}^{M} N_d}\right) \tag{2.7}$$

In the test set D_{test}, M is the number documents and N_d is the number of words, $p(w_d)$ is the probability assigned to the word w_d by the topic model, and N_d is the number of words in document d.

2.3.3.2 Topic Coherence: UCI and UMass

Topic coherence measures the semantic similarity of high-scoring tokens within the clusters. This is meant to provide insight into how well a topic can be interpreted. The higher the scores, the more coherence is assigned to the topic. Both UCI and UMass calculate topic coherence by summing up pairwise distributional similarity scores across the group of topic words (Stevens et al., 2012):

$$coherence(V) = \sum((v_i, v_j) \in V)()score(v_i, v_j, \epsilon) \tag{2.8}$$

Here, V represents a topic generated by the model and ϵ is added for smoothing, i.e. to avoid division by zero.

UCI assigns a score to a word pair based on their pointwise mutual information (PMI). The probabilities of words are determined by counting how often they appear together in a moving window across an external corpus, which, in comparison to established semantic evaluations, can be regarded as a more external benchmark (Stevens et al., 2012):

$$UCI(v_i, v_j, \epsilon) = \log \frac{p(v_i, v_j) + \epsilon}{p(v_i)p(v_j)} \tag{2.9}$$

Here, $p(v_i, v_j)$ is the probability of the words v_i and v_j occurring together in the test corpus, $p(v_i)$ and $p(v_j)$ are the individual probabilities of the words v_i and v_j appearing in the text corpus.

The UMass metric is as score that takes into account document co-occurrence. It possesses an intrinsic quality, aiming to validate whether the models have effectively learned about the data present within the corpus itself. UMass calculates counts using the initial training corpus of the topic models (Stevens et al., 2012):

$$UMass(v_i, v_j, \epsilon) = \log \frac{D(v_i, v_j) + \epsilon}{D(v_j)} \tag{2.10}$$

In this formula, $D(v_i, v_j)$ is the number of documents that containing the words v_i and v_j, while $D(v_j)$ counts the number of documents that contain v_j.

2.3.3.3 BERTopic

BERTopic, introduced by Grootendorst (2022), uses BERT embeddings and a class-based variation of TF-IDF (c-TF-IDF) to create topic clusters. It functions as follows:

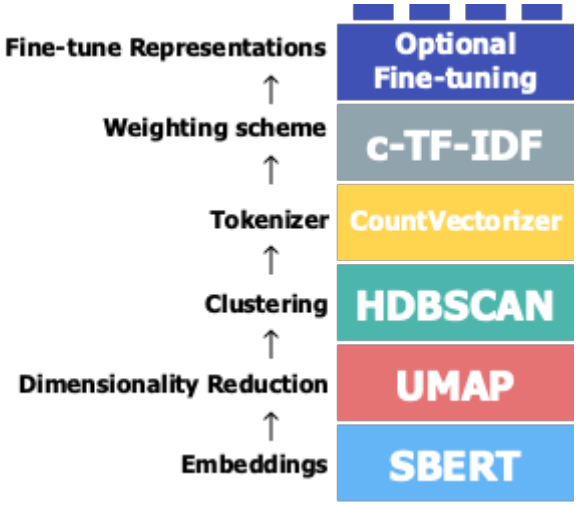


Figure 1: Visualisation of the BERTopic algorithm by (Grootendorst, 2022)

.

1. Each document in the text corpus that is analysed is encoded into embeddings using Sentence-BERT.
2. All embeddings are reduced in dimensionality by applying Uniform Manifold Approximation and Projection (UMAP) (McInnes et al., 2018). This allows for better clustering.
3. The reduced vectors are clustered via Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN), which is able to find dense data clusters of all shapes and sizes.

4. The original documents are mapped to the clusters and each cluster is assigned a topic using a modified version using c-TF-IDF. This entails weighing words against clusters, i.e. it measures the importance of each word within its respective cluster.

5. Finally, the number of topics can be reduced by merging the c-TF-IDF scores of the least common topic with the one that is most similar to it.

BERTopic is a highly flexible framework and has been shown to produce easily interpretable topics from text corpora. It allows for adjustments in all of the steps mentioned above. For instance, one can easily implement different embedding models in the first step, or apply another method to reduce the vector dimensionality in the second step. One may also use the topics generated and process them even further, e.g. by engineering a prompt for a large language model (LLM) that should label them, as BERTopic does not output labels for the topics (Grootendorst, 2022).

3 Method

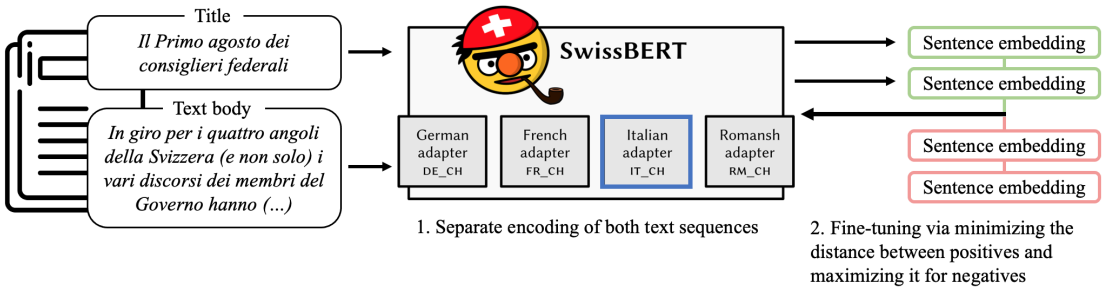


Figure 2: Visualisation of the supervised SimCSE training approach.

3.1 Fine-tuning

The complete fine-tuning script can be accessed via Github.[1]

To fine-tune SwissBERT for sentence embeddings, a self-supervised SimCSE approach without hard negatives was chosen. Analogous to the original SwissBERT, Swiss news articles serve as the training data for this. The documents are split into sequence pairs, where one sequence consists of the article's title and – if available – its lead concatenated, while the other contains the text body (see Figure 2). This can be regarded as a self-supervised learning training approach (Liu et al., 2023). The title-body pairs represent (h_i, h_i^+) in the constrastive loss training objective 2.1.

3.1.1 Dataset

The fine-tuning data consists of over 1.5 million Swiss news articles obtained through the Swissdox@LiRI database[2] in German, French, Italian, and Romansh (see Table 1). In order to avoid data leakage from the test set (see section 3.2.1.1, all articles published in the medium *20 Minuten* and *20 minutes* (incl. online) were filtered

[1] https://github.com/jgrosjean-mathesis/sentence-swissbert/tree/main/training
[2] https://swissdox.linguistik.uzh.ch/

out. While all German and French articles selected from the corpus were published between 2020 and 2023, the publication dates of the Italian and Romansh media range from 2000 to 2023, due to the scarcity of recent publications in these languages. All of the news articles were pre-processed in the same way as SwissBERT's original training data, namely by removing html markup elements and marked-up author and photographer names as well as separating layout elements using the special token </s>(Vamvas et al., 2023).

Language	**Documents**	**Tokens**
German	760 350	621 107 750
French	644 416	567 688 406
Italian	63 666	35 109 282
Romansh	39 732	16 376 397
Total	**1 508 414**	**1 240 281 835**

Table 1: Composition of the dataset used for fine-tuning SwissBERT. The number of documents and tokens in the four languages is reported.

Additionally, the title, lead, and text body from each article were extracted and entered into various tsv-files separated by language. The language code was added to the respective file names of the training data batches. This was crucial for the fine-tuning process of SwissBERT, since the language adapter has to be switched according to the language of the input data. Hence, it needs to be specified as a separate parameter when the fine-tuning data is opened. The pre-processing base script can be accessed on Github.[3]

The structure of the SimCSE train script provided by Gao et al. (2021)[3] was updated and adapted according to the SwissBERT architecture. This included modifying several components in order to make it compatible with current versions of the transformers library and Python. Furthermore, it was necessary to add a separate X-MOD model architecture configuration, since the given script was only applicable to BERT and RoBERTa models. Here, a part of the relevant code snippet is shown:

```
class XmodForCL(XmodPreTrainedModel):
    _keys_to_ignore_on_load_missing = [r"position_ids"]

    def __init__(self, config, *model_args, **model_kargs):
        super().__init__(config)
        self.model_args = model_kargs["model_args"]
        self.roberta = XmodModel(config, add_pooling_layer=False)
        self.init_weights()
```

[3] https://github.com/jgrosjean-mathesis/sentence-swissbert/blob/main/preprocessing/swissdox_new.py

[3] https://github.com/princeton-nlp/SimCSE

```
9
10          cl_init(self, config)
```

Additionally, the fine-tuning script had to be extended by adding a language switch component, so that the model would continuously adjust its adapter according to the training data language during the fine-tuning process. This was tackled by having the script consecutively open a set of training data batch files as opposed to just one single file. These batches were randomly ordered. Each contained data in one language only, which was specified in the file name. Hence, the script was able to identify the language of each training data set and adjusted the used language adapter accordingly, using the given function *set_default_language()*:

```
for dataset_batch in dataset_batches:
    if "de" in dataset_batch:
        model.set_default_language("de_CH")
        print("setting default language to de_CH for",
    dataset_batch)
    elif "fr" in dataset_batch:
        model.set_default_language("fr_CH")
        print("setting default language to fr_CH for",
    dataset_batch)
    elif "it" in dataset_batch:
        model.set_default_language("it_CH")
        print("setting default language to it_CH for",
    dataset_batch)
    elif "rm" in dataset_batch:
        model.set_default_language("rm_CH")
        print("setting default language to rm_CH for",
    dataset_batch)
```

During fine-tuning on SimCSE, the language adapters were frozen via the function included in the X-MOD configuration *freeze_language_adapters()*:

```
logger.info("Freezing adapters")
for layer in model.roberta.encoder.layer:
    if layer.output.adapter_layer_norm is not None:
        for parameter in layer.output.adapter_layer_norm.parameters
    ():
            parameter.requires_grad = False
    for parameter in layer.output.adapter_modules.parameters():
        parameter.requires_grad = False
```

All the other parameters were updated, including the input embeddings. This is in line with the original method SwissBERT was trained with (Vamvas et al., 2023).

3.1.2 Hyperparameters

The training data was padded / truncated to 512 tokens, so that it fits the input limit. The model was fine-tuned in one single epoch, using a learning rate of 1e-5 and the AdamW optimizer (Loshchilov and Hutter, 2019), a batch size of 512 and a temperature of 0.05, which is suggested by the creators of SimCSE Gao et al. (2021). The model was updated based on MEAN pooling, following the findings mentioned in section 2.2.2. In order to not run into an error by overloading the cache available, the parameter *per_device_train_batch_size* was set to 4 and *gradient_accumulation_steps* to 128, so that not too many computational ressources were used at the same time while still being in line with the batch size 512.

3.2 Evaluation

Commonly used evaluation toolkits for sentence embeddings like SentEval (Conneau and Kiela, 2018) are limited to non-Swiss languages and an international context. Thus, the newly fine-tuned SentenceSwissBERT model is evaluated on three custom, Switzerland-related NLP tasks in German, French, Italian, and Romansh. It is measured against the original SwissBERT and a multilingual Sentence-BERT model that showed the strongest performance in the first two evaluation tasks (see appendix A.2).

3.2.1 Datasets

For the evaluation tasks, two different datasets are used.

3.2.1.1 20 Minuten

For the first two evaluation tasks, the *20 Minuten* dataset is used (Kew et al., 2023). It comprises articles published in *20 Minuten*, one of the most widely circulated German-language newspapers in Switzerland. The printed version is handed out for free, e.g. at bus stops and train stations. The articles tend to be relatively short and cover a variety of topics. Most of the documents in the dataset include a short article summary and topic tags.

Given its format and features, the 20 Minuten dataset is especially suitable for assessing SentenceSwissBERT's performance. It includes the right format (news

Task	Language	Documents
Document retrieval	German	499
	French	499
	Italian	499
	Romansh	499
Text classification: train set	German	4 986
Text classification: test set	German	1 240
	French	1 240
	Italian	1 240
	Romansh	1 240

Table 2: Composition of the documents in the 20 Minuten dataset that were used for the first two evaluation tasks.

Category	Train articles	Test articles
accident	244	60
corona	1 468	367
economy	768	192
film	247	61
football	627	156
germany	250	62
social media	288	71
switzerland	300	743
ucraine war	268	66
usa	526	131
Total	**4 986**	**1 240**

Table 3: Composition of the train and test sets of the text classification task in the 20 Minuten corpus, including the respective counts per category.

articles), is tied to a Swiss context, and the summary and topic tag components can be leveraged for semantic-similarity-related NLP tasks. All articles present in the 20 Minuten corpus were removed from the original fine-tuning data in all languages, so that no data leakage can occur.

In order to expand the evaluation to French, Italian, and Romansh, the relevant parts of the articles were machine-translated using the Google Cloud API (FR, IT) and the Textshuttle API (RM). Using machine translation allows for a controlled comparison across languages when evaluating, since all documents share the same structure and content. Moreover, manual annotations can be automatically projected to the other languages without a need for additional annotation. A potential downside of machine translation is that the distribution of the test data does not reflect the

diversity of human-written text. Tables 2 and 3 show statistics of the data used for evaluation.

3.2.1.2 Swiss Broadcasting Corporation (SRG SSR)

For the third evaluation task, a separate dataset is consolidated. It contains data from one year (April 2023 - March 2024) published by the Swiss Broadcasting Corporation (SRG SSR), which is the official public broadcasting agency of Switzerland. It includes the following language-specific media channels:

- SRF (Schweizer Radio und Fernsehen)[3]
- RTS (Radio Télévision Suisse)[3]
- RSI (Radiotelevisione svizzera di lingua italiana)[3]
- RTR (Radiotelevisiun Svizra Rumantscha)[3]

While originally television channels, all of these also publish news articles on their respective websites, which is where the data is taken from. These media lend themselves for assessing SentenceSwissBERT's performance, as they are published in one of the four national languages respectively and the publications are all related to Switzerland. Table 4 shows the exact composition of this evaluation dataset.

Task	Language	Documents
Topic modeling	SRF (German)	26 998
	RTS (French)	12 569
	RSI (Italian)	14 292
	RTR (Romansh)	6 731

Table 4: Composition of the documents in SRG SSR dataset that was used for the topic modeling evaluation tasks.

3.2.2 Baseline Models

3.2.2.1 SwissBERT

Albeit not specifically trained for this, sentence embeddings can already be extracted from the last hidden layer of the original SwissBERT encoder model via MEAN pool-

[3] https://srf.ch
[3] https://rts.ch
[3] https://rsi.ch
[3] https://rtr.ch

ing. The input language is specified, just like in its newly fine-tuned version. This comparison demonstrates whether there is value in fine-tuning the model specifically for sentence embeddings.

3.2.2.2 Sentence-BERT

Reimers and Gurevych (2019) propose several multilingual sentence embedding models for semantic similarity tasks.[3] In this work, the *distiluse-base-multilingual-cased-v1* model is opted for as a baseline, as it shows the strongest performance for the given evaluation tasks (see Appendix A.2). It has originally been trained following the multilingual knowledge distillation approach introduced in Section 2.2.5, using the multilingual Universal Sentence Encoder (Yang et al., 2020). This version of Sentence-BERT supports various languages, among them French, German, and Italian, but not Romansh. Unlike with SwissBERT, the input language does not need to be specified. This model has a similar number of parameters as SwissBERT (see Table 5). However, it maps to a 512-dimensional embedding space and, hence, is computationally more efficient than SwissBERT.

Model	**Vocabulary**	**Parameters**
Sentence-BERT	119 547	135 127 808
SwissBERT	50 262	160 101 888

Table 5: Vocabulary sizes and parameter counts of the two baseline models. The fine-tuned SentenceSwissBERT has the same size as the original model.

The other multilingual Sentence-Transformer *paraphrase-multilingual-mpnet-base-v2* suggested for semantic similarity tasks is much larger (278 043 648 parameters). Although this model maps to a 768-dimensional space, similar to SwissBERT, it performed worse than *distiluse-base-multilingual-cased-v1* in the evaluation tasks (see Appendix A.2). Thus, it was disregarded.

3.2.3 Tasks

The complete Python scripts for all evaluation tasks are available on Github.[4]

[3] https://www.sbert.net/examples/training/multilingual/README.html

[4] https://github.com/jgrosjean-mathesis/sentence-swissbert/tree/main/evaluation

3.2.3.1 Document Retrieval

Following the same approach as Palangi et al. (2016), the embedding of each article's summary is compared to the article's content embedding and then matched by choosing the pair with the highest cosine similarity score. Since in this retrieval task, each query only has one correct match, it is not possible to report a precision, recall, or F1-score. Hence, the performance is reported simply via the top-1 accuracy score, which is solely based on how many summaries were matched with the correct content in relation to the total number of articles processed:

$$\text{Accuracy} = \frac{\text{Correct Matches}}{\text{All Matches}} \tag{3.1}$$

The relevant code snippet where this is implemented looks as follows:

```
predicted_matches = {}

for summary_id, summary_embedding in summary_embeddings.items():
    max_similarity = -1
    predicted_text_id = None
    cosine_score_dict = {}

    for text_id, test_embedding in text_embeddings.items():
        cosine_score = util.cos_sim(summary_embedding,
   text_embedding)
        cosine_score_dict[content_id] = cosine_score

    predicted_text_id, max_cosine_score = max(cosine_score_dict.
   items(), key=lambda x: x[1])
    predicted_matches[summary_id] = predicted_text_id
```

Given its setup, there is no train-test split necessary for this task, so none was conducted. It is performed monolingually (where the summary is written in the same language as the article) and cross-lingually, which means the experiment assesses the model's capabilities in both contexts. The experiment covers each language pair possible in the set of German, French, Italian, and Romansh.

3.2.3.2 Text Classification

In this task, certain topic tags already present in the 20 Minuten dataset are manually mapped to ten categories. All documents without these (or overlapping) chosen topic tags are excluded and disregarded in this experiment. Then, a random train-test split with a 80/20 ratio is performed once on the remaining data for every

category respectively. The exact number of files per category as well as the chosen categories are displayed in Table 3. Next, the text classification is carried out utilizing a k-nearest neighbor approach (see section 2.3.2.2): The text body of each test article is compared to every embedding from the training data via cosine similarity. Subsequently, the topic tag of its one nearest neighbor from the training set (highest similarity) is assigned to it. The relevant code snippet were the classification is carried out looks as follows:

```
test_ids_with_predicted_category = {}

for test_id, test_embedding in test_ids_with_embeddings.items():
    max_similarity = -1
    predicted_category = None
    cosine_score_dict = {}

    for train_embedding, train_category in
   train_categories_with_embeddings.items():
    cosine_score = util.cos_sim(test_embedding, train_embedding)
    cosine_score_dict[cosine_score] = train_category

    max_cosine_score = max(cosine_score_dict.keys())
    predicted_category = cosine_score_dict[max_cosine_score]

    test_ids_with_predicted_category[test_id] = predicted_category
```

To assess cross-lingual transfer, the training data is kept in German for the assessment of each of the four languages, while the test data is machine-translated to French, Italian and Romansh. As the categories vary in frequency, the weighted average of all categories' F1-scores is reported (see section 2.3.2). Due to the nature of this experiment, there is no possibility to perform a k-fold cross validation.

3.2.3.3 Topic Modeling

This larger down-stream task involves a BERTopic algorithm that makes use of SentenceSwissBERT's embeddings. It uses the second evaluation dataset (see section 3.2.1.2), which is comprised of Swiss news articles published by SRF (German), RTS (French), RSI (Italian), and RTR (Romansh). Replacing the model that outputs the embeddings with any other encoder model is suggested by the creator of BERTopic (Grootendorst, 2022). The algorithm is applied once with the baseline and once with the new fine-tuned model, so that their performances can be compared. The original SwissBERT is disregarded for this task.

The topics generated are assessed in terms of perplexity, to assess the model's capa-

bilities of generalizing, as well as UCI and UMass to check whether the outputted topics are coherent. Furthermore, the topic's coherence is assessed in a qualitative way, additionally taking into account human intepretability, topic diversity, and other noteworthy analysis points.

3.2.3.3.1 Code Implementation The BERTopic algorithm entails five steps introduced in section 2.3.3.3. These can all be manually adjusted according to the experiment. The crucial part in the underlying experiment is the first step, i.e. choosing a sentence embedding model. As the chosen baseline is a Sentence-BERT encoder (distiluse-base-multilingual-cased-v1), it can easily be specified, as Sentence-BERT is suggested as the default encoder model for BERTopic. However, to set up sentence embeddings via SentenceSwissBERT, which requires the specification of the input language, a manual embedding function that encodes the input documents and passes them on to the second step in the right format was necessary.

Additionally, the given function *reduce_frequent_words()* is activated during clustering. This should avoid stop words from appearing in the topics that are outputted. This approach is especially useful in the underlying case, as there exist no stop word lists for Romansh. Also, as BERTopic allows an easy way to do this, the number of topics that are identified is set to a maximum of 20 and the words per topic to 15. This was done in order to streamline the outputs across all media and simplify their comparison.

BERTopic does not offer a built-in function for evaluation metrics. As the idea for this experiment is to calculate perplexity, UCI, and UMass for each sub-corpus, the computation of these had to be implemented manually. As for perplexity, BERTopic assigns probabilities for each topic to every document it processes. Using the numpy package, these can be used as a base to calculate the perplexity score:

```
log_perplexity = -1 * np.mean(np.log(np.sum(probabilities, axis=1))
    )
perplexity = np.exp(log_perplexity)
```

UCI and UMass were calculated automatically by making use of the class *CoherenceModel* by Gensim, which is a Python package especially designed for topic modeling (Rehurek and Sojka, 2010). The calculation of these was implemented adjusting a code[5] suggested directly by the creators of BERTopic:

```
def calculate_coherence(topic_model, topics, documents):
    """calculates topic coherence scores for assessment"""
```

[5] https://github.com/MaartenGr/BERTopic/issues/90

```
    documents = pd.DataFrame({"Document": documents, "ID": range(len(documents)), "Topic": topics})
    documents_per_topic = documents.groupby(['Topic'], as_index=False).agg({'Document': ' '.join})
    cleaned_docs = topic_model._preprocess_text(documents_per_topic.Document.values)

    vectorizer = topic_model.vectorizer_model
    analyzer = vectorizer.build_analyzer()
    tokens = [analyzer(doc) for doc in cleaned_docs]
    dictionary = corpora.Dictionary(tokens)
    corpus = [dictionary.doc2bow(token) for token in tokens]
    topic_words = [[words for words, _ in topic_model.get_topic(topic)] for topic in range(len(set(topics))-1)]

    umass_coherence_model = CoherenceModel(topics=topic_words, texts=tokens, corpus=corpus, dictionary=dictionary, coherence='u_mass')
    umass_coherence = umass_coherence_model.get_coherence()

    uci_coherence_model = CoherenceModel(topics=topic_words, texts=tokens, corpus=corpus, dictionary=dictionary, coherence='c_uci')
    uci_coherence = uci_coherence_model.get_coherence()

    return umass_coherence, uci_coherence
```

4 Results

4.1 Document Retrieval

All the results for the document retrieval evaluation task are reported in Table 6.

Encoder Model	**Summary Language**	**Article Language**			
		de-ch	fr-ch	it-ch	rm-ch
SwissBERT	de-ch	87.20%	78.36%	72.95%	40.68%
	fr-ch	86.52%	84.97%	78.96%	40.84%
	it-ch	83.17%	80.17%	84.17%	33.41%
	rm-ch	46.08%	39.10%	43.39%	83.17%
Sentence-BERT	de-ch	91.80%	90.98%	**90.38**%	62.53%
	fr-ch	90.78%	93.19%	90.78%	63.36%
	it-ch	88.12%	**91.29**%	91.58%	65.71%
	rm-ch	70.59%	73.48%	73.55%	73.35%
SentenceSwissBERT	de-ch	**93.40**%	**92.79**%	90.18%	**91.58**%
	fr-ch	**94.33**%	**93.99**%	**90.98**%	**90.07**%
	it-ch	**92.08**%	90.85%	**92.18**%	**88.50**%
	rm-ch	**92.16**%	**89.44**%	**88.43**%	**91.58**%

Table 6: Results of the document retrieval evaluation task using the 20 Minuten dataset (Kew et al., 2023). The top-1 accuracy score is reported. The best results per language pair are marked in bold print.

Despite not being specifically trained for tasks that make use of sentence embeddings, the SwissBERT base model achieves good results in all of the monolingual retrieval tasks, always attaining a top-1 accuracy score of over 80% in them. The cross-lingual retrieval tasks between German, French, and Italian also functioned decently well, with them all achieving a score over 70%. However, the performance clearly declines in all the cross-lingual tasks that involve the processing of Romansh

text, where the original SwissBERT never obtains an accuracy of over 50%.

SentenceSwissBERT outperforms its base model SwissBERT in all of the 16 experiments conducted. This shows a clear improvement in comparison to the original model. The largest difference can be noted in the processing of Romansh text, where it beats the strongest baseline it is compared to by almost 30%. Out of all three models the experiments were conducted with, the original SwissBERT model achieves the worst results across almost all retrieval tasks. Only when it comes to the monolingual retrieval in Romansh does it achieve the second-best performance, placing before Sentence-BERT.

The Sentence-BERT model distiluse-base-multilingual-cased-v1 was able to achieve scores of over 88% in all monolingual and cross-lingual retrieval tasks for the languages it was trained in, i.e. German, French, and Italian. It obtained the best score among all models looked at in this research in two of the retrieval experiments, namely when using Italian summaries for retrieving French texts and using German summaries to retrieve Italian texts. Although the Sentence-BERT model has not been trained on Romansh text data (zero-shot setting), it performed better than the SwissBERT base model (trained in Romansh) in all the cross-lingual tasks involving Romansh. It can also be noted that this zero-shot retrieval works better when the summary is in Romansh and the texts it is compared to are in the familiar languages, as opposed to the other way around. For instance, when using a Romansh summary to retrieve a text in French, the accuracy is over 10% higher than when the languages are reversed.

The fine-tuned SentenceSwissBERT obtains better results than both baselines in all document retrieval experiments except for two. With 88.50% being the weakest score, the results can be considered very good across the board. It outperforms its base model in all document retrieval tasks. Compared to the chosen Sentence-BERT model, it performs better by 10.07% on average, which is significant. The clearest difference can be seen with Romansh text data, which it has specifically been trained on. This shows that SentenceSwissBERT's embeddings would be a useful tool for similar retrieval tasks.

4.2 Text Classification

Table 7 presents the results of the text classification evaluation task.

The original SwissBERT model achieves decent results when classifying German texts, even outperforming Sentence-BERT in this case. However, especially when it

Encoder Model	**Training Language**	**Test Language**			
		de-ch	fr-ch	it-ch	rm-ch
SwissBERT	de-ch	77.93%	69.62%	67.09%	43.79%
Sentence-BERT	de-ch	77.23%	76.83%	**76.90%**	65.35%
SentenceSwissBERT	de-ch	**78.49%**	**77.18%**	76.65%	**77.20%**

Table 7: Results of the cross-lingual text evaluation task using the 20 Minuten dataset (Kew et al., 2023). The weighted F1-score is reported and the best results are marked in bold print.

comes to classifying Romansh text strings, the performance declines, just like in the document retrieval task.

Sentence-BERT shows promising results for the monolingual and cross-lingual evaluation, achieving relatively high scores for all the languages it has been trained in. It scored the best out of all the models looked at when classifying Italian texts. Again, despite this version of Sentence-BERT not being trained in Romansh, it was still able to score above 65% for this language.

SentenceSwissBERT improves over the baselines in this task as well, never scoring below 75%. It scores the best in all text classification experiments except for the Italian one, where the Sentence-BERT model is slightly more accurate. However, it outperforms this strong baseline by 3.3% on average. Again, the most drastic difference can be seen in the Romansh experiment, showing a difference in weighted F1-score of over 11%. This shows another potential case in which SentenceSwissBERT could be of use.

4.3 Topic Modeling

4.3.1 Quantitative Metrics

Table 8 shows the evaluation metric scores perplexity, UCI, and UMass for the topic modeling down-stream task.

Encoder model	**Medium**	**Perplexity**	**UCI**	**UMass**
SentenceBERT	SRF (de)	1.81	-1.53	-0.40
	RTS (fr)	**1.78**	-1.44	-0.42
	RSI (it)	1.84	-0.90	-0.40
	RTR (rm)	1.73	-2.03	**-0.46**
SentenceSwissBERT	SRF (de)	**1.77**	**-0.11**	**-0.34**
	RTS (fr)	1.87	**-0.40**	**-0.27**
	RSI (it)	**1.76**	**-0.28**	**-0.35**
	RTR (rm)	**1.68**	**-1.61**	-0.52

Table 8: Results of the topic modeling downstream task using the SRG SSR dataset. Perplexity, UCI, and UMass are reported. Each best score per medium is marked in bold print.

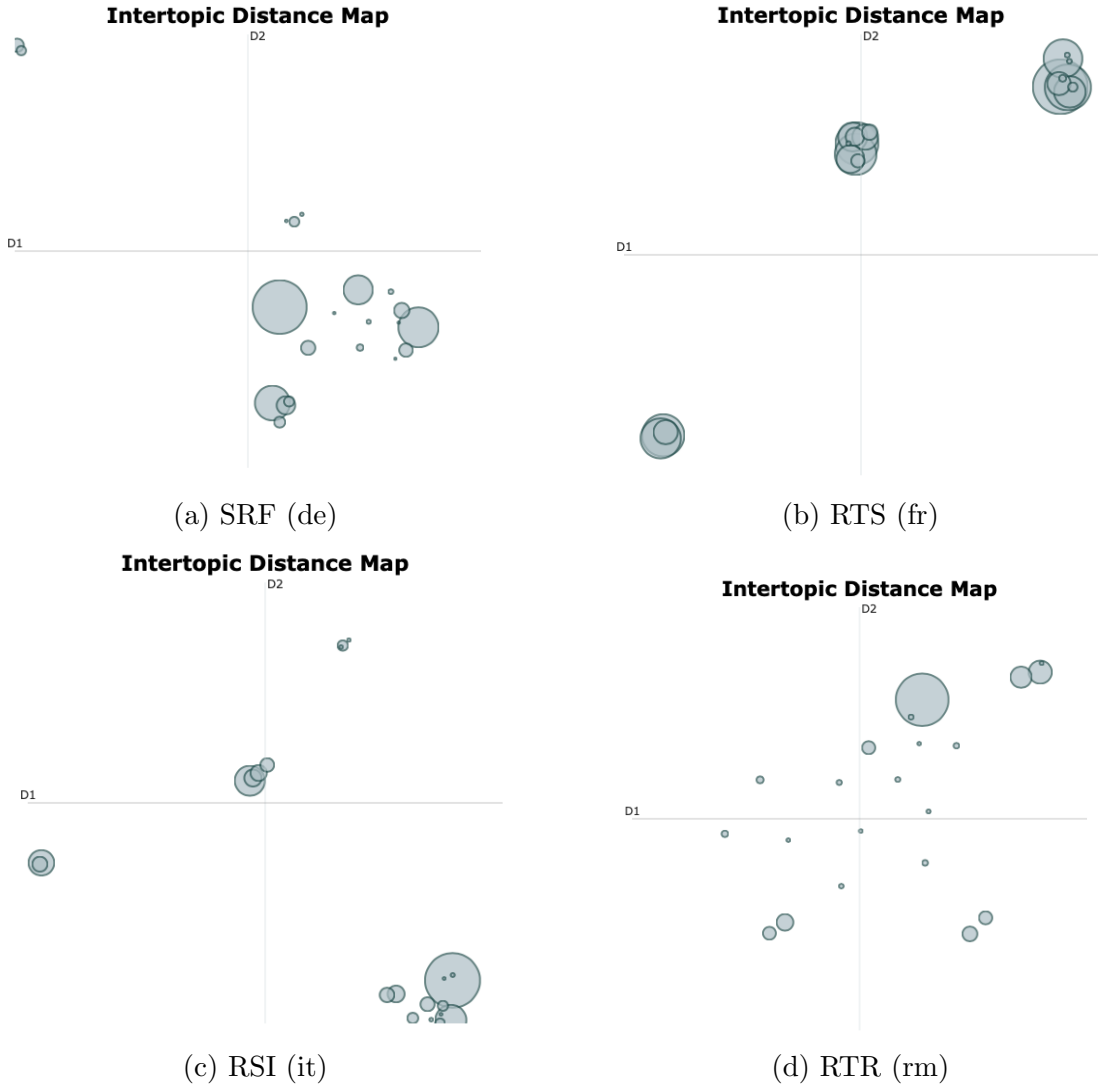
Both models achieve decent perplexity scores. SentenceSwissBERT scores better in German, Italian, and Romansh, while Sentence-BERT outperforms the other model when it comes to French topics.

Based on the UCI and Umass scores, Sentence-BERT and SentenceSwissBERT both generate coherent topics via the BERTopic algorithm. SentenceSwissBERT achieves better scores all over, except for the UMass coherence in the Romansh topics. The difference is especially visible when looking at UCI, where there is an average difference of 0.88 between the two models' scores.

4.3.2 Topic Visualisations

BERTopic allows for an easy 2D visualization of the generated topics via the built-in function *visualize_topics()*. Their output is what is called the *Intertopic Distance Map*, which looks similar LDAvis, introduced by (Sievert and Shirley, 2014). This entails the c-TF-IDF vectors of each topic being mapped to a two-dimensional space via UMAP. It is attempted to leverage these visual representations shown in figures

3 and 4 in order to draw conclusions on how heterogeneous the topics output by both models are.



(a) SRF (de)

(b) RTS (fr)

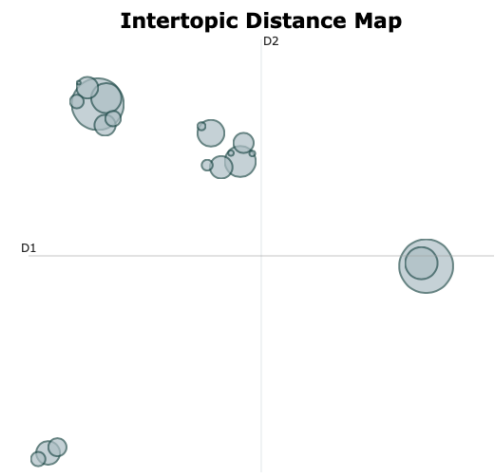
(c) RSI (it)

(d) RTR (rm)

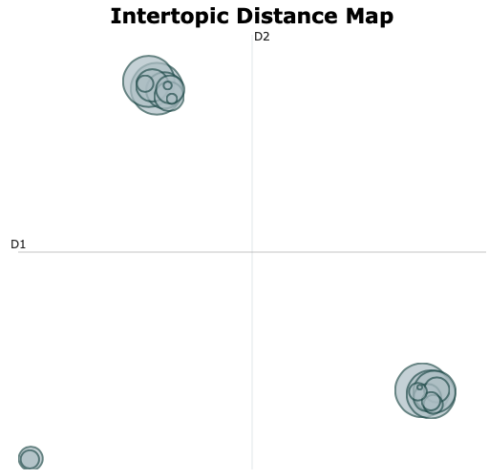
Figure 3: The intertopic distance maps based on the topics generated via Sentence-BERT are displayed.

The topic clusters generated via Sentence-BERT embeddings are fairly spread out, which is interpreted as there being high diverstiy among topics. The only exception is the French corpus (RTS), which shows three clear clusters.

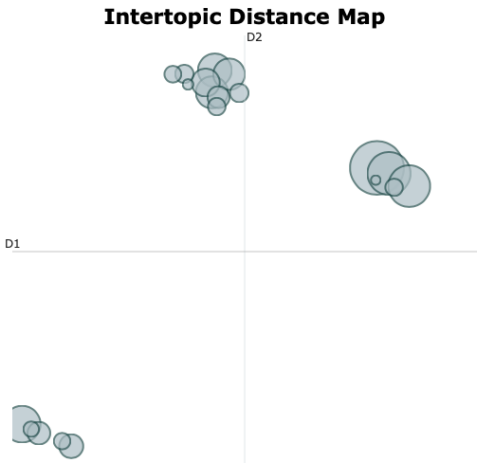
In SentenceSwissBERT's topic maps, clear clusters can be recognized in all four languages. There are fewer small bubbles seen in between than there are in the topic maps of Sentence-BERT.



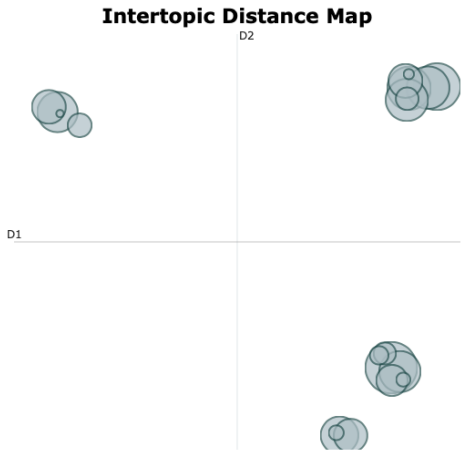
(a) SRF (de)



(b) RTS (fr)



(c) RSI (it)



(d) RTR (rm)

Figure 4: The intertopic distance maps based on the topics generated via SentenceSwissBERT are displayed.

4.3.3 Qualitative Analysis

All topics generated in this task can be seen in the appendix A.3.

Tables 9, 10, 11, and 12 all show examples of one coherent, easily interpretable topic that was identified across the whole quadrilingual corpus by both models, namely the armed conflicts in Ukraine and the Middle East. This allows for a direct comparison of the topics output using both models.

The topics generated based on the SRF data (Table 9) are largely similar, with both topics sharing 9 words.

Medium	Sentence-BERT		SentenceSwissBERT	
SRF (de)	ukraine:	0.27	quellen:	0.29
	russischen:	0.25	ukraine:	0.28
	2023:	0.25	russischen:	0.26
	reuters:	0.24	informationen:	0.26
	quellen:	0.23	sind:	0.24
	ukrainische:	0.23	kriegsparteien:	0.24
	krieg:	0.22	ukrainische:	0.23
	israel:	0.22	israel:	0.23
	russland:	0.22	2023:	0.23
	sind:	0.21	ort:	0.22
	kiew:	0.21	osten:	0.22
	informationen:	0.21	krieg:	0.22
	soldaten:	0.21	russland:	0.22
	ukrainischen:	0.2	nahen:	0.22
	hamas:	0.2	gazastreifen:	0.21

Table 9: Two similar topics incl. word probabilities identified in the German subcorpus about the armed conflicts in Ukraine and the Middle East.

Medium	Sentence-BERT		SentenceSwissBERT	
RTS (fr)	ukraine:	0.29	guerre:	0.23
	russie	0.26	pays	0.22
	russe	0.26	gaza	0.21
	russes	0.22	hamas	0.2
	guerre	0.22	isral	0.2
	poutine	0.21	larme	0.2
	migrants	0.2	russie	0.19
	pays	0.2	lukraine	0.19
	vladimir	0.2	lonu	0.18
	asile	0.19	que	0.18
	turquie	0.19	russe	0.18
	moscou	0.19	au	0.18
	président	0.18	chine	0.18
	erdogan	0.18	ont	0.18
	ukrainien	0.18	dans	0.18

Table 10: Two similar topics incl. word probabilities identified in the French subcorpus about the armed conflict in Ukraine and, in the case of SentenceSwissBERT, the Middle East.

Medium	**Sentence-BERT**		**SentenceSwissBERT**	
RSI (it)	ucraina:	0.3	russia:	0.23
	russia	0.27	mosca	0.21
	kiev	0.26	presidente	0.21
	mosca	0.24	kiev	0.21
	russo	0.23	guerra	0.2
	putin	0.23	ha	0.2
	ucraino	0.22	gaza	0.2
	presidente	0.22	russo	0.2
	guerra	0.22	putin	0.2
	russi	0.21	israele	0.19
	cina	0.21	ucraina	0.19
	zelensky	0.21	militare	0.19
	russa	0.2	forze	0.19
	wagner	0.2	stati	0.19
	ha	0.2	della	0.19

Table 11: Two similar topics incl. word probabilities identified in the Italian subcorpus about the armed conflicts in Ukraine and, in the case of SentenceSwissBERT, the Middle East.

In the clusters assembled from the French corpus (Table 10), we see a clearer difference. While Sentence-BERT outputted a more unified topic that only includes words in relation to the war in Ukraine, SentenceSwissBERT outputted one that also includes the Middle Eastern conflict, making it more diverse. The SentenceSwissBERT topic also contains a considerable amount of words that do not contribute to the topic per se, namely *que*, *au*, *ont*, and *dans*. It is also noteworthy that the nouns that start in a vowel in SentenceSwissBERT's topic are preceded by their definitive article *l'*. However, they are missing the apostrophe, e.g. *lukraine* and *lonu*, which makes them orthographically incorrect.

The topics generated from the Italian news articles (Table 10) also overlap by a great deal (9 identical words). Here, SentenceSwissBERT also includes the Middle East topic and Sentence-BERT does not.

The Romansh topics show the same distinction as the French and Italian ones, i.e. Sentence-BERT outputting a more unified topic that only includes terms relating to the war in Ukraine, as seen in Table 12. Similar to what happened in French, SentenceSwissBERT also adds the definite article *l'* but omits the apostrophe, e.g. *lucraina*, *larmada* and *lonu*.

Medium	**Sentence-BERT**		**SentenceSwissBERT**	
RTR (rm)	ucraina:	0.35	lucraina:	0.26
	russia	0.32	gaza	0.26
	russa	0.26	strivla	0.26
	ucranais	0.25	lisrael	0.24
	president	0.24	russia	0.24
	russ	0.23	hamas	0.24
	selenski	0.23	larmada	0.22
	guerra	0.23	guerra	0.22
	armas	0.23	israeliana	0.21
	ucranaisa	0.23	sajan	0.2
	putin	0.23	attatgas	0.2
	pajais	0.22	stadis	0.2
	moscau	0.22	tenor	0.2
	haja	0.21	lonu	0.2
	russas	0.21	ha	0.2

Table 12: Two similar topics incl. word probabilities identified in the Romansh sub-corpus about the armed conflicts in Ukraine and, in the case of SentenceSwissBERT, the Middle East.

Tables 13 and 14 each show easily interpretable, individual topics that each were not identified by the other model respectively. For these cases, it can be noted that Sentence-BERT was able to identify more broad, internationally relevant clusters, while SentenceSwissBERT was capable of filtering out topics that are more regionally relevant to the country of Switzerland.

The left-hand topic in table 13 shows a topic that is of global relevance; it is mainly related to East Asian (especially Chinese) politics. The right-hand topic, on the other hand, is particularly related to Switzerland. It describes the bankruptcy of the Swiss bank *Credit Suisse*. It can be noted that many Switzerland-specific terms are listed, such as *credit*, *suisse*, *cs*, *snb*, *franken*, *finma*, and *bund*.

Table 14 again paints the picture of Sentence-BERT identifying a more general, universally-relevant topic, this time concerning beekeeping and pollination. SentenceSwissBERT, on the other hand, was able to cluster a topic about competitive skiing, which can be considered a local topic especially relevant in Switzerland. Noteworthy is the inclusion of particular alpine skiing disciplines, such as *superg*, *descente*, and *slalom*, as well as the mention of proper names of professional Swiss skiers, like *marco*, *odermatt* (Marco Odermatt) and *gutbehrami* (Lara Gut-Behrami).

Medium	Sentence-BERT		SentenceSwissBERT	
srf (de)	china	0.49	ubs	0.51
	taiwan	0.4	credit	0.45
	südkorea	0.32	cs	0.45
	chinesische	0.32	suisse	0.43
	chinas	0.3	bank	0.4
	peking	0.3	milliarden	0.32
	nordkorea	0.3	bankenkrise	0.32
	chinesischen	0.29	banken	0.31
	xi	0.28	übernahme	0.31
	usa	0.27	grossbank	0.3
	indien	0.26	snb	0.28
	beziehungen	0.24	nationalbank	0.28
	li	0.23	franken	0.27
	taiwans	0.22	finma	0.27
	kim	0.22	bund	0.25

Table 13: A topic related to East Asian politics that was only identified by Sentence-BERT and a Switzerland-specific topic that was only identified by SentenceSwissBERT, both from the German sub-corpus.

Medium	Sentence-BERT		SentenceSwissBERT	
RTS (fr)	abeilles	1.05	odermatt	0.37
	miel	0.61	coupe	0.37
	ruches	0.58	podium	0.33
	colonies	0.55	superg	0.33
	apiculteurs	0.51	2e	0.33
	sucre	0.41	monde	0.32
	apiculture	0.4	tape	0.32
	abeille	0.38	discipline	0.31
	mellifères	0.38	gutbehrami	0.31
	biodiversité	0.38	3e	0.31
	roundup	0.37	descente	0.31
	pollinisateurs	0.36	marco	0.3
	disparition	0.36	médaille	0.3
	sauvages	0.35	slalom	0.3
	miels	0.34	4e	0.3

Table 14: A topic related to beekeeping/biodiversity that was only identified by Sentence-BERT and a Switzerland-specific topic that was only identified by SentenceSwissBERT, both from the French sub-corpus.

Medium	**Sentence-BERT**		**SentenceSwissBERT**	
SRF (de)	ich	0.23	prozent	0.19
	oder	0.23	das	0.19
	sie	0.21	ist	0.19
	ist	0.2	dass	0.19
	es	0.2	ubs	0.19
	man	0.2	für	0.19
	nicht	0.2	eine	0.18
	kann	0.2	es	0.18
	wenn	0.2	sie	0.18
	auch	0.2	nicht	0.18
	eine	0.2	auch	0.18
	dass	0.2	zu	0.18
	zu	0.19	cs	0.18
	das	0.19	sp	0.18
	werden	0.19	von	0.17

Table 15: Two examples for non-interpretable topics from the German sub-corpus.

Tables 15 and 16 show topics identified by both models in the German and French corpus that are not interpretable, as they mostly include words that serve a grammatical function as opposed to holding much semantic value. The left-hand topic in table 16 could loosely be interpreted as having something to do with climate change, as it contains the words *eau* (water), *éléctricité* (electricity), *degrés* (degrees), and *climatique*, albeit most of the topic consists of function words, i.e. articles, conjunctions, prepositions, and pronouns. This is even more extreme in the right-hand topic, where 12 out of the 15 words are functional only.

Medium	Sentence-BERT		SentenceSwissBERT	
RTS (fr)	eau	0.22	on	0.19
	loup	0.2	plus	0.19
	loups	0.19	sont	0.19
	plus	0.19	ou	0.19
	électricité	0.18	les	0.19
	les	0.18	est	0.18
	degrés	0.18	dans	0.18
	des	0.18	pour	0.18
	sont	0.18	en	0.18
	en	0.18	pas	0.18
	climatique	0.18	une	0.18
	est	0.18	que	0.18
	dans	0.18	des	0.18
	pour	0.18	suisse	0.18
	on	0.18	et	0.18

Table 16: Two examples for non-interpretable topics from the French sub-corpus.

4.4 Model Publication

In order to make the fine-tuned SentenceSwissBERT publicly available, it was uploaded to the HuggingFace Transformers library.[1] Accompanying the model, a model card was set up to elaborate on the fine-tuning method, linking it to the SimCSE method applied. The following example code was included to specify how SentenceSwissBERT is used to extract embeddings and then applied to calculate the cosine similarity of two sample sentences in German and French:

```
import torch
from transformers import AutoModel, AutoTokenizer
from sklearn.metrics.pairwise import cosine_similarity

# Load SentenceSwissBERT model
model_name = "jgrosjean-mathesis/sentence-swissbert"
model = AutoModel.from_pretrained(model_name)
tokenizer = AutoTokenizer.from_pretrained(model_name)

def generate_sentence_embedding(sentence, language):

    # Set adapter to specified language
```

[1] https://huggingface.co/jgrosjean-mathesis/sentence-swissbert

```
    if "de" in language:
        model.set_default_language("de_CH")
    if "fr" in language:
        model.set_default_language("fr_CH")
    if "it" in language:
        model.set_default_language("it_CH")
    if "rm" in language:
        model.set_default_language("rm_CH")

    # Tokenize input sentence
    inputs = tokenizer(sentence, padding=True, truncation=True, return_tensors="pt", max_length=512)

    # Take tokenized input and pass it through the model
    with torch.no_grad():
        outputs = model(**inputs)

    # Extract sentence embeddings via mean pooling
    token_embeddings = outputs.last_hidden_state
    attention_mask = inputs['attention_mask'].unsqueeze(-1).expand(token_embeddings.size()).float()
    sum_embeddings = torch.sum(token_embeddings * attention_mask, 1)
    sum_mask = torch.clamp(attention_mask.sum(1), min=1e-9)
    embedding = sum_embeddings / sum_mask

    return embedding

# Define two sentences
sentence_1 = ["Der Zug kommt um 9 Uhr in Zuerich an."]
sentence_2 = ["Le train arrive a Lausanne a 9h."]

# Compute embedding for both
embedding_1 = generate_sentence_embedding(sentence_1, language="de")
embedding_2 = generate_sentence_embedding(sentence_2, language="fr")

# Compute cosine similarity
cosine_score = cosine_similarity(embedding_1, embedding_2)

# Output the score
print("The cosine score for", sentence_1, "and", sentence_2, "is", cosine_score)
```

Output:

```
The cosine score for ['Der Zug kommt um 9 Uhr in Zuerich an.']
and ['Le train arrive a Lausanne a 9h.'] is [[0.85555995]]
```

Furthermore, the two initial evaluation tasks were described and their results documented in the model description. These provide insights into potential usages of the model in NLP tasks. i.e. how SentenceSwissBERT is intended to be used. In addition, a small paragraph on risk, bias, and limitations was added, serving as a caveat for users.

In order to display the model's functionality in an even more tangible way, a very simple user interface was set up using Gradio and made available on a Hugging Face space.[2] The link is also automatically referenced on the model card itself. In the space, users can specify one source sentence and three target sentences. By clicking on *Submit*, the cosine similarity scores between the source sentence and the three target sentences are calculated and ranked. This functionality is inspired by the built-in Semantic Similarity Inference API, offered by Hugging Face, which is not applicable with the model. It mimics a simple search function (see figure 5).

The complete code for the Gradio interface is accessible on the respective Hugging Face space.[3]

Finally, the first part of this paper was accepted for the Swiss text analytics conference *SwissText 2024* and made accessible online (Grosjean and Vamvas, 2024).

[2] https://huggingface.co/spaces/jgrosjean/SentenceSwissBERT

[3] https://huggingface.co/spaces/jgrosjean/SentenceSwissBERT/blob/main/app.py

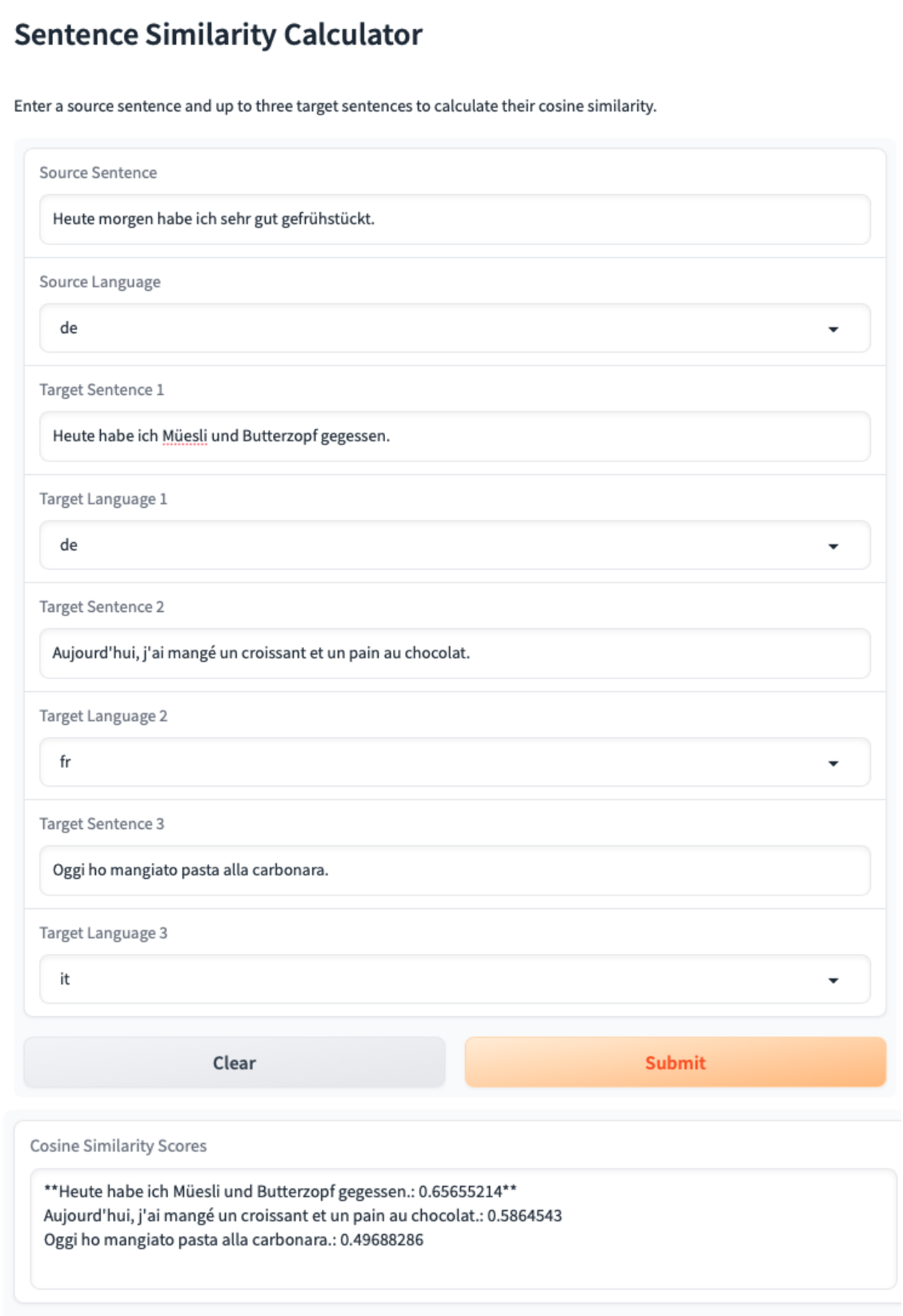


Figure 5: Gradio interface demonstrating a search function that utilizes SentenceSwissBERT's embeddings.

5 Discussion

5.1 Discussion

In this section, all results of the three evaluation tasks are discussed and interpreted.

5.1.1 Document Retrieval

The results in the retrieval tasks show that embeddings from the original SwissBERT can already be used for this type of task, at least as long as everything is kept in the same language (monolingual text processing). However, especially when the cross-lingual tasks involve Romansh text, its performance is not optimal.

It is noted that the Sentence-BERT model distiluse-base-multilingual-cased-v1 does a fantastic job at retrieving documents in both a monolingual and cross-lingual context. Concerning cross-lingual retrieval, the model probably benefits from its training approach, where it was ensured that sentences with the same meaning in two different languages (translation pairs) are mapped close to each other in the embedding space. In the case of Romansh, this Sentence-BERT model even proves to be useful in a zero-shot setting, achieving decent results. This might be due to the fact that its training data includes other Latin languages with similar structures and vocabulary to Romansh, namely French, Italian, Portuguese, and Spanish. What is interesting to see is that the zero-shot retrieval using this model functions better when the summary is in the unfamiliar language and the document to be searched for is in a language known to the model, compared to vice versa. This might be due to the summaries being shorter and, thus, their format being closer to the original training data. It is concluded that, in a (partial) zero-shot setting, the distiluse-base-multilingual-cased-v1 model is better at embedding sentences and short text sequences than longer ones.

The SwissBERT model fine-tuned for sentence embeddings is able to outperform both baseline models in 14 out of 16 cases. This not only proves the fine-tuning of the original model to be useful, but also shows that the new model performs

better than a general sentence embedding model like Sentence-BERT in this task. SentenceSwissBERT's high scores can be attributed to the fact that the way this document retrieval task was set up mirrors the way in which SentenceSwissBERT was fine-tuned: On one side, there are short text sequences, whether it is the concatenated title and lead of an article or a text summary. On the other side, there are longer text sequences that the short sequences are succeeded by. In both contexts, the shorter sequence's function is to describe or summarize the longer sequence's content, i.e. they are semantically similar.

The only two experiments where Sentence-BERT outperforms the fine-tuned SwissBERT model both are cross-lingual tasks that include Italian. The number of Italian and Romansh articles in the dataset used for fine-tuning SentenceSwissBERT was considerably smaller than was used for German in French. This might explain why, despite achieving decent results, SentenceSwissBERT's performance in Italian is not always up to par with the Sentence-BERT baseline. Even though the dataset was also limited in terms of Romansh data, the new model still does a great deal better than Sentence-BERT here. This is not surprising, since the latter was not trained in that language at all. Nevertheless, we see that for all experiments that involve Romansh, SentenceSwissBERT's scores are slightly worse than for the rest of the languages; with 3 out of 7 of the scores being below 90%.

5.1.2 Text Classification

The results of these experiment again show that, although its use is limited, the original SwissBERT model's embeddings can already be employed for this type of NLP task. It even outperforms Sentence-BERT in the German classification, which is probably owing to its original training data and the evaluation dataset having the same format.

Sentence-BERT neither has this advantage, nor has it been explicitly trained on Switzerland-related text data. Considering this, it does extremely well in the German, French, and Italian classification, further proving that its embeddings provide a strong basis for semantic-based text processing. The zero-shot setting for Romansh works surprisingly well, which again shows that the model, despite not being familiar with the language, probably benefits from being trained in Latin languages related to Romansh.

SentenceSwissBERT again achieves the best average results, outperforming both baselines in 3 out of 4 experiments. Again, it can be observed that the performance in Italian is just shy of achieving as high of a score as Sentence-BERT, which further

supports the point made in section 5.1.1, mentioning that the fine-tuning dataset is lacking in Italian data. However, SentenceSwissBERT still attains the best results overall, with none of its scores falling below 76%.

Overall, while the test classification F1-scores are decent, there is potential for improvement. None of the models manage to score above 80%. This might be due to there being 10 categories. The higher the number of categories, the more difficult the classification task is for the model. Although articles with overlapping categories were filtered out from the dataset, there might still be topic overlaps when it comes to the actual content of the article. Additionally, the tags in the dataset sometimes seem to be put arbitrarily and do not follow a fixed set of rules. For instance, the category *germany* includes all texts related to the country of Germany. This does make for a more heterogenous class rather than a clear-cut category like *football* or *corona*, where there is a specific vocabulary associated with the topic. All of this does not facilitate a clear classification, even for humans.

5.1.3 Topic Modeling

This section discusses all results attained from the topic modeling evaluation task comparing Sentence-BERT and SentenceSwissBERT to set up topics using the BERTopic approach. Both models can be deemed suitable to be used with BERTopic, which makes sense, as they are both transformer-based encoder models that BERTopic is designed for to be used with.

5.1.3.1 Quantitative Metrics

Both Sentence-BERT and SentenceSwissBERT are able to obtain good perplexity scores, although SentenceSwissBERT seems to do minimally better. This may indicate that SentenceSwissBERT is better at capturing topics and generalizing based on these.

SentenceSwissBERT achieves better topic coherence, especially in terms of the UCI metric (average improvement of 0.88), which means that the topics this model outputted probably reflect the underlying themes of the corpus it was given better. However, this has to be taken with a grain of salt, as the differences in the scores are relatively small. It can also be deducted from the qualitative analysis that both models mostly produce easily interpretable, coherent topics.

5.1.3.2 Qualitative Analysis

Looking at the topics themselves, the results show mostly coherent topics that seem to accurately represent common themes among the articles in the corpus. Both models identify large topic clusters, like the armed conflicts in Ukraine and the Middle East, the political happenings in the US, or even the recent technical advancements in AI (see A.3 for all topics outputted). Although most of the topics make sense and seem well-assembled, there are also cases for topics with little to no interpretable content, showing that the algorithm is still flawed. These topics could probably be filtered out by feeding the BERTopic code with stop word lists, which was refrained from doing in the underlying experiment. What stands out is that SentenceSwissBERT's embeddings seem to especially do well at assembling topics that are regionally relevant to Switzerland, such as national elections and competitive skiing. Additionally, the words that represent the topics in French and Romansh sometimes are incorrectly tokenized, which is never the case for Sentence-BERT. This might have to do with SwissBERT's utilizing a different tokenization system than Sentence-BERT.

6 Conclusion

Regarding the results, it can be concluded that the chosen fine-tuning approach for enhancing SwissBERT to set up semantic sentence embeddings was successful. Hence, contrastive learning, more exactly the SimCSE approach with title–body pairs, is an effective fine-tuning approach for a masked language model. It can be concluded that a title (and a lead), in the context of a news article, can be used as a representation for its text body. This is plausible, as the functionality of a title and a lead is to summarize (or tease) the content they accompany.

Using just a subset of 1.5 million articles from the original pre-training dataset, a clear improvement on the two first sentence-level evaluation tasks has been achieved compared to the baseline, in both mono- and cross-lingual settings. This is outstanding, considering the fact that the fine-tuning was carried out without any multilingual data, such as translations, i.e. it did not require resources that might be difficult to acquire. It is striking that SentenceSwissBERT, in the context of these evaluation tasks, performs mostly equally or better than other models that have been trained on significantly larger datasets and via more granular, supervised training, such as the Sentence-BERT model distiluse-base-multilingual-cased-v1. This indicates that combining the modular architecture of SwissBERT with the contrastive learning framework SimCSE is effective. SentenceSwissBERT's ability for cross-lingual transfer learning is remarkable. This experiment showcases the new model's potential to be used in scenarios where data resources are limited, e.g. to one language only.

What especially stands out among the results is SentenceSwissBERT's high performance when processing German, French, and Romansh text. The high scores in German can be attributed to the linguistic differences between Swiss Standard German and Standard German, as well as the Swiss context, e.g. Swiss-specific names, toponyms, helvetisms. With regards to Romansh, it is plausible that the model yields better results than Sentence-BERT, as it has specifically been trained on Romansh text data, and, to the author's knowledge, is the first transformer-based model to be specifically trained on this type of task in Romansh.

The newly fine-tuned SentenceSwissBERT is proposed to be employed in the context of processing Switzerland-related text via sentence or document embeddings across the four Swiss national languages, in particular for news articles. It has been shown to compute high-quality sentence embeddings that can be used for a variety of tasks, mono- and cross-lingual. The models' cross-lingual transfer capabilities are especially interesting for resource-constrained scenarios. Future work could also explore whether including training data from other domains could further improve the generality of the model.

References

Aly Abdelrazek, Yomna Eid, Eman Gawish, Walaa Medhat, and Ahmed Hassan Yousef. 2022. Topic modeling algorithms and applications: A survey. *Information Systems*, 112:102131.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder. *CoRR*, abs/1803.11175.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ran Ding, Ramesh Nallapati, and Bing Xiang. 2018. Coherence-aware neural topic modeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 830–836, Brussels, Belgium. Association for Computational Linguistics.

Hongchao Fang and Pengtao Xie. 2020. CERT: contrastive self-supervised learning for language understanding. *CoRR*, abs/2005.12766.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Maarten R. Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *ArXiv*, abs/2203.05794.

Juri Grosjean and Jannis Vamvas. 2024. Fine-tuning the swissbert encoder model for embedding sentences and documents.

Dan Jurafsky and James H. Martin. 2024. Speech and language processing (3rd ed. draft). Release date: February 3, 2024.

Tannon Kew, Marek Kostrzewa, and Sarah Ebling. 2023. 20 minuten: A multi-task news summarisation dataset for german. In *SwissText 2023: 8th Swiss Text Analytics Conference*.

Xiao Liu, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang. 2023. Self-supervised learning: Generative or contrastive. *IEEE Transactions on Knowledge and Data Engineering*, 35(1):857–876.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. 2018. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861.

Hamid Palangi, Li Deng, Yelong Shen, Jianfeng Gao, Xiaodong He, Jianshu Chen, Xinying Song, and Rabab Ward. 2016. Deep sentence embedding using long short-term memory networks: analysis and application to information retrieval. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 24(4):694–707.

Christian Samuel Perone, Roberto Silveira, and Thomas S. Paula. 2018. Evaluation of sentence embeddings in downstream and linguistic probing tasks. *ArXiv*, abs/1806.06259.

Jonas Pfeiffer, Naman Goyal, Xi Lin, Xian Li, James Cross, Sebastian Riedel, and Mikel Artetxe. 2022. Lifting the curse of multilinguality by pre-training modular transformers. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3479–3495, Seattle, United States. Association for Computational Linguistics.

Radim Rehurek and Petr Sojka. 2010. Software framework for topic modelling with large corpora.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.

M Rüdiger, D Antons, AM Joshi, and TO Salge. 2022. Topic modeling revisited: New evidence on algorithm performance and quality metrics. *PloS one*, 17(4):e0266325.

Carson Sievert and Kenneth Shirley. 2014. LDAvis: A method for visualizing and interpreting topics. In *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*, pages 63–70, Baltimore, Maryland, USA. Association for Computational Linguistics.

Keith Stevens, Philip Kegelmeyer, David Andrzejewski, and David Buttler. 2012. Exploring topic coherence over many models and many topics. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 952–961, Jeju Island, Korea. Association for Computational Linguistics.

Jannis Vamvas, Johannes Graën, and Rico Sennrich. 2023. SwissBERT: The multilingual language model for Switzerland. In *Proceedings of the 8th edition of*

the Swiss Text Analytics Conference, pages 54–69, Neuchatel, Switzerland. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2020. Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online. Association for Computational Linguistics.

Senhui Zhang, Tao Ji, Wendi Ji, and Xiaoling Wang. 2022. Zero-shot event detection based on ordered contrastive learning and prompt-based prediction. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2572–2580, Seattle, United States. Association for Computational Linguistics.

Declaration of Independent Authorship

Original work

I expressly declare that the written work I submitted to the University of Zurich in the spring/autumn semester of 2024 with the title

Fine-tuning the SwissBERT Encoder Model for Embedding Sentences and Documents

is an original work written by myself, in my own words, and without unauthorized assistance. If it is a work by several authors, I confirm that the relevant parts of the work are correctly and clearly marked and can be clearly assigned to the respective author.

I also confirm that the work has not been submitted in whole or in part to receive credit for another module at the University of Zurich or another educational institution, nor will it be submitted in the future.

Use of sources

I expressly declare that I have identified all references to external sources (including tables, graphics etc.) contained in the above work as such. In particular, I confirm that, without exception and to the best of my knowledge, I have indicated the authorship both for verbatim statements (citations) and for statements by other authors reproduced in my own words (paraphrases).

Use of text generation models

I expressly declare that I have not only identified existing external sources, but also any automatically generated text that is contained in the above work. I have used the same citation style as if the text had been generated by a human to indicate the source of the automatically generated text. If the contribution of text generation models cannot be linked to specific text passages (see the associated guidelines), I have included a chapter describing the contributions of the text generation model. I acknowledge that no explicit citation is necessary where text generation models are merely used correctively (to improve grammar or idiomaticity of my own words).

Sanctions

I acknowledge that a thesis that is used to acquire credit and proves to be plagiarism with the meaning of the document *Erläuterung des Begriffs „Plagiat“* leads to a grade

deduction in minor cases, a grade 1 (one) in more severe cases, without the possibility of revision, and in very severe cases can have the corresponding legal and disciplinary consequences according to §§ 7ff of the “Disziplinarordnung der Universität Zürich” and § 36 of the “Rahmenordnung für das Studium in den Bachelor- und Master-Studiengängen der Philosophischen Fakultät der Universität Zürich”.

I confirm with my signature that this information is correct:

Name: Grosjean First Name: Juri

Matriculation number: 15-562-382

Date: 31. Mai 2024 Signature: 

A Appendix

A.1 Pre-training Dataset Media Composition

lematin.ch	99 939	fr
24heures.ch	73 385	fr
tdg.ch	69 498	fr
Le Temps	63 130	fr
24 heures	62 004	fr
Tribune de Genève	57 604	fr
blick.ch	51 556	de
rsi.ch	51 526	it
letemps.ch	48 353	fr
rts.ch	47 397	fr
cash.ch	46 750	de
blick.ch	43 178	fr
rtr.ch	39 732	rm
srf.ch	29 536	de
nzz.ch	28 091	de
tagblatt.ch	27 279	de
luzernerzeitung.ch	23 855	de
Aargauer Zeitung / MLZ	21 868	de
Neue Zürcher Zeitung	18 408	de
Le Matin Dimanche	18 352	fr
Thurgauer Zeitung	17 335	de
Blick	14 636	de
landbote.ch	13 089	de
Tages-Anzeiger	13 040	de
bazonline.ch	12 709	de
aargauerzeitung.ch	12 309	de
bernerzeitung.ch	12 207	de
Zofinger Tagblatt / MLZ	11 888	de

tagesanzeiger.ch	11 612	de
berneroberlaender.ch	11 603	de
thunertagblatt.ch	11 581	de
zsz.ch	11 517	de
L'Illustré	11 231	fr
langenthalertagblatt.ch	11 184	de
zuonline.ch	11 120	de
Basler Zeitung	10 895	de
derbund.ch	10 748	de
schweizer-illustrierte.ch	10 620	de
Zuger Zeitung	10 557	de
bz - Zeitung für die Region Basel	10 528	de
handelszeitung.ch	9 790	de
pme.ch	9 491	fr
Der Bund	9 396	de
Werdenberger & Obertoggenburger	9 214	de
Der Landbote	9 122	de
Zürichsee-Zeitung	9 019	de
fuw.ch	8 791	de
Luzerner Zeitung	8 651	de
Badener Tagblatt	8 435	de
Urner Zeitung	8 284	de
St. Galler Tagblatt	8 117	de
Wiler Zeitung	8 003	de
Berner Zeitung	7 777	de
Appenzeller Zeitung	7 548	de
Zürcher Unterländer	7 425	de
Oltner Tagblatt / MLZ	7 420	de
badenertagblatt.ch	7 140	de
Berner Oberländer	7 138	de
Femina	7 106	fr
Toggenburger Tagblatt	7 032	de
Thuner Tagblatt	6 982	de
solothurnerzeitung.ch	6 120	de
bzbasel.ch	5 921	de
RTS.ch	5 914	fr
Obwaldner Zeitung	5 854	de
Nidwaldner Zeitung	5 844	de
TV 8	5 677	fr

Sonntagsblick	5 606	de
Grenchner Tagblatt	5 530	de
Solothurner Zeitung / MLZ	5 450	de
BZ - Langenthaler Tagblatt	5 277	de
SonntagsZeitung	5 228	de
Limmattaler Zeitung / MLZ	5 042	de
NZZ am Sonntag	4 991	de
Finanz und Wirtschaft	4 962	de
SWI swissinfo.ch	4 855	it
Glückspost	4 621	de
Limmattaler Zeitung	4 513	de
limmattalerzeitung.ch	4 488	de
rts Vidéo	4 092	fr
Die Weltwoche	4 011	de
Bilan	3 979	fr
oltnertagblatt.ch	3 958	de
grenchnertagblatt.ch	3 857	de
swissinfo.ch	3 575	it
www.swissinfo.ch	3 541	it
swissinfo.ch	3 525	fr
PME Magazine	3 244	fr
illustre.ch	3 077	fr
Schweizer Illustrierte	3 068	de
Handelszeitung	2 917	de
srf Video	2 558	de
Die Wochenzeitung	1 953	de
bellevue.nzz.ch	1 919	de
Thalwiler Anzeiger/Sihltaler	1 826	de
Zuger Presse	1 781	de
HZ Insurance	1 617	de
Schweizer Familie	1 570	de
weltwoche.ch	1 466	de
Beobachter	1 446	de
Zugerbieter	1 409	de
Guide TV Cinéma	1 384	fr
weltwoche.de	1 267	de
Tele	1 176	de
Bilanz	1 085	de
swissinfo.ch	1 004	de

encore!	986	fr
Beobachter.ch	984	de
Das Magazin	982	de
züritipp (Tages-Anzeiger)	882	de
NZZ am Sonntag Magazin	823	de
TV Star	764	de
weltwoche-daily.ch	719	de
bilanz.ch	596	de
SWI swissinfo.ch	587	fr
Streaming	535	de
HZ Insurance	529	fr
NZZ PRO Global	446	de
Schweizer LandLiebe	441	de
glueckspost.ch	399	de
encore! (dt)	274	de
Newsnet / 24 heures	227	fr
TV Land & Lüt	215	de
NZZ Geschichte	151	de
SI Sport	143	de
Newsnet / Berner Zeitung	143	de
Bolero	142	de
boleromagazin.ch	118	de
NZZ Folio	109	de
beobachter.ch	107	de
Aargauer Zeitung / MLZ	91	fr
HZ Insurance	77	it
SI Gruen	70	de
L'Illustré Sport	70	fr
Newsnet / Basler Zeitung	69	de
Newsnet / Der Bund	58	de
Bolero F	56	fr
Schweiz am Wochenende	47	fr
Badener Tagblatt	34	fr
Schweizer Versicherung	31	fr
Newsnet / Le Matin	28	fr
Newsnet / Tribune de Genève	25	fr
Schweizer Illustrierte Style	23	it
Grenchner Tagblatt	22	fr
Oltner Tagblatt / MLZ	21	fr

Werdenberger & Obertoggenburger	21	it
Solothurner Zeitung / MLZ	20	fr
Limmattaler Zeitung / MLZ	20	fr
Finanz und Wirtschaft	18	fr
NZZ Online	16	de
Schweizer Versicherung	16	it
TV4	12	de
Limmattaler Zeitung	9	fr
rts Video	7	fr
SWI swissinfo.ch	6	de
Newsnet / Tages-Anzeiger	6	de
Handelszeitung	6	it
Berner Oberländer	5	fr
Thuner Tagblatt	5	fr
berneroberlaender.ch	4	fr
Beobachter.ch	4	it
thunertagblatt.ch	3	fr
Neue Zürcher Zeitung	3	it
cash.ch	2	fr
Blick	2	it
Berner Zeitung	2	it
srf.ch	2	it
weltwoche.de	2	it
Blick	1	fr
bernerzeitung.ch	1	fr
fuw.ch	1	fr
Sonntagsblick	1	fr
Basler Zeitung	1	fr
weltwoche.ch	1	fr
weltwoche.de	1	fr
srf.ch	1	fr
bazonline.ch	1	fr
rtr.ch	1	it
derbund.ch	1	it
St. Galler Tagblatt	1	it
Die Weltwoche	1	it
Das Magazin	1	it
nzz.ch	1	it
Basler Zeitung	1	it

Schweiz am Sonntag / MLZ	1	it
blick.ch	1	it
Cash	1	it
bazonline.ch	1	it

Table 16: Composition of the dataset used to fine-tune the SwissBERT model according to medium and language.

A.2 Evaluation Results of Sentence-BERT Baselines

Encoder Model	**Summary Language**	**Article Language**			
		de-ch	fr-ch	it-ch	rm-ch
paraphrase-multilingual-mpnet-base-v2	de-ch	75.01%	81.76%	79.56%	18.44%
	fr-ch	75.18%	83.57%	81.56%	19.87%
	it-ch	72.28%	78.87%	79.56%	19.25%
	rm-ch	53.64%	53.91%	57.11%	19.44%
distiluse-base-multilingual-cased-v1	de-ch	**91.80%**	**90.98%**	**90.38%**	**62.53%**
	fr-ch	**90.78%**	**93.19%**	**90.78%**	**63.36%**
	it-ch	**88.12%**	**91.29%**	**91.58%**	**65.71%**
	rm-ch	**70.59%**	**73.48%**	**73.55%**	**73.35%**

Table 17: Results of the document retrieval evaluation task using two multilingual Sentence-BERT models. The top-1 accuracy score is reported. The best results per language pair are marked in bold print.

Encoder Model	**Training Language**	**Test Language**			
		de-ch	fr-ch	it-ch	rm-ch
paraphrase-multilingual-mpnet-base-v2	de-ch	75.42%	75.64%	73.88%	39.38%
distiluse-base-multilingual-cased-v1	de-ch	**77.23%**	**76.83%**	**76.90%**	**65.35%**

Table 18: Results of the cross-lingual text evaluation task using the two multilingual Sentence-BERT models. A weighted F1-score is reported and the best results are marked in bold print.

A.3 Complete Topic Modeling Results

Numbering	Sentence-BERT		SentenceSwissBERT	
Topic 0				
0	league	0.24	gegen	0.27
1	gegen	0.23	league	0.27
2	sieg	0.21	minute	0.24
3	wm	0.21	resultate	0.23
4	nach	0.21	nach	0.22
5	resultate	0.2	spiel	0.22
6	saison	0.2	partie	0.22
7	den	0.2	fc	0.21
8	jährige	0.19	bersicht	0.21
9	minute	0.19	mit	0.21
10	dem	0.19	den	0.2
11	mit	0.19	sieg	0.2
12	spiel	0.19	saison	0.2
13	im	0.19	tabelle	0.2
14	schweizer	0.19	zum	0.2
Topic 1				
0	prozent	0.19	quellen	0.29
1	das	0.19	ukraine	0.28
2	ist	0.19	russischen	0.26
3	dass	0.19	informationen	0.26
4	ubs	0.19	sind	0.24
5	für	0.19	kriegsparteien	0.24
6	eine	0.18	ukrainische	0.23
7	es	0.18	israel	0.23
8	sie	0.18	2023	0.23
9	nicht	0.18	ort	0.22
10	auch	0.18	osten	0.22
11	zu	0.18	krieg	0.22
12	cs	0.18	russland	0.22
13	sp	0.18	nahen	0.22
14	von	0.17	gazastreifen	0.21
Topic 2				
0	ukraine	0.27	rennen	0.3
1	russischen	0.25	sekunden	0.29

2	2023	0.25	rang	0.28
3	reuters	0.24	podest	0.24
4	quellen	0.23	platz	0.24
5	ukrainische	0.23	weltcup	0.23
6	krieg	0.22	schweizer	0.23
7	israel	0.22	am	0.23
8	russland	0.22	km	0.22
9	sind	0.21	verstappen	0.22
10	kiew	0.21	saison	0.22
11	informationen	0.21	im	0.22
12	soldaten	0.21	auf	0.21
13	ukrainischen	0.2	frauen	0.21
14	hamas	0.2	dem	0.21
Topic 3				
0	am	0.21	ich	0.23
1	bis	0.2	oder	0.23
2	grad	0.19	sie	0.21
3	es	0.19	ist	0.2
4	sbb	0.19	es	0.2
5	ein	0.18	man	0.2
6	auf	0.18	nicht	0.2
7	des	0.18	kann	0.2
8	wurden	0.18	wenn	0.2
9	von	0.18	auch	0.2
10	zu	0.18	eine	0.2
11	brienz	0.18	dass	0.2
12	werden	0.18	zu	0.19
13	auch	0.18	das	0.19
14	dem	0.18	werden	0.19
Topic 4				
0	erdogan	0.27	trump	0.3
1	partei	0.24	regierung	0.22
2	regierung	0.23	donald	0.22
3	türkei	0.23	biden	0.21
4	kilicdaroglu	0.21	er	0.2
5	land	0.2	erdogan	0.2
6	parlament	0.2	partei	0.2
7	migranten	0.19	kirche	0.2

8	milei	0.19	republikaner	0.2
9	eu	0.19	zu	0.19
10	er	0.19	als	0.18
11	italien	0.19	hat	0.18
12	opposition	0.19	das	0.18
13	parteien	0.18	dass	0.18
14	für	0.18	fr	0.18
Topic 5				
0	tiere	0.23	the	0.22
1	wölfe	0.22	er	0.21
2	rudel	0.2	musik	0.2
3	oder	0.2	als	0.2
4	wolf	0.2	sie	0.19
5	ist	0.19	film	0.19
6	werden	0.19	mit	0.19
7	das	0.19	ist	0.19
8	auch	0.19	ein	0.19
9	man	0.19	das	0.19
10	nicht	0.19	und	0.19
11	ein	0.19	ich	0.19
12	sie	0.19	ihr	0.18
13	es	0.19	song	0.18
14	bauern	0.19	von	0.18
Topic 6				
0	co2	0.32	brienz	0.24
1	strom	0.3	dorf	0.24
2	energie	0.26	polizei	0.23
3	klimaschutz	0.24	behörden	0.23
4	gesetz	0.24	feuer	0.23
5	gas	0.22	wurden	0.22
6	werden	0.21	schäden	0.22
7	rösti	0.21	menschen	0.22
8	schweiz	0.2	am	0.22
9	für	0.2	erdbeben	0.22
10	anlagen	0.2	worden	0.21
11	prozent	0.2	des	0.2
12	solaranlagen	0.2	sagte	0.2
13	das	0.2	häuser	0.2

14	energien	0.2	sei	0.2
Topic 7				
0	ki	0.38	prozent	0.26
1	daten	0.3	franken	0.23
2	musk	0.29	initiative	0.23
3	intelligenz	0.27	ja	0.22
4	mond	0.26	fr	0.21
5	google	0.24	vorlage	0.2
6	nasa	0.24	das	0.2
7	chatgpt	0.24	stimmen	0.2
8	unternehmen	0.23	bundesrat	0.19
9	elon	0.23	dass	0.19
10	microsoft	0.23	werden	0.19
11	twitter	0.22	eine	0.19
12	sunrise	0.22	mehr	0.19
13	tiktok	0.22	kantone	0.19
14	erde	0.21	kanton	0.19
Topic 8				
0	charles	0.31	sp	0.36
1	musik	0.3	svp	0.34
2	könig	0.29	fdp	0.33
3	song	0.26	grünen	0.31
4	iii	0.23	wahlen	0.3
5	buss	0.23	sitz	0.3
6	swift	0.23	mitte	0.3
7	tia	0.23	ständerat	0.29
8	turner	0.23	wahlgang	0.29
9	esc	0.23	nationalrat	0.28
10	songs	0.22	partei	0.27
11	krönung	0.22	jans	0.27
12	prinz	0.22	sitze	0.26
13	album	0.22	überset	0.26
14	er	0.21	bundesrat	0.26
Topic 9				
0	trump	0.66	staatsanwaltschaft	0.25
1	donald	0.45	daten	0.22
2	republikaner	0.42	rubiales	0.22
3	biden	0.41	habe	0.21

4	us	0.38	des	0.2
5	trumps	0.35	dass	0.2
6	vorwahlen	0.33	brian	0.2
7	joe	0.31	er	0.2
8	desantis	0.31	fall	0.19
9	mccarthy	0.31	nicht	0.19
10	anklage	0.3	urteil	0.19
11	demokraten	0.3	puk	0.19
12	usa	0.3	das	0.19
13	präsidenten	0.29	von	0.19
14	präsident	0.28	sei	0.19
Topic 10				
0	china	0.49	ki	0.26
1	taiwan	0.4	strom	0.25
2	südkorea	0.32	intelligenz	0.23
3	chinesische	0.32	werden	0.22
4	chinas	0.3	energie	0.21
5	peking	0.3	ist	0.2
6	nordkorea	0.3	das	0.2
7	chinesischen	0.29	wir	0.2
8	xi	0.28	chatgpt	0.19
9	usa	0.27	wie	0.19
10	indien	0.26	so	0.19
11	beziehungen	0.24	fr	0.19
12	li	0.23	auch	0.19
13	taiwans	0.22	dass	0.19
14	kim	0.22	co	0.19
Topic 11				
0	ich	0.41	grad	0.39
1	guten	0.34	schnee	0.32
2	abend	0.31	temperaturen	0.31
3	kann	0.28	meteo	0.29
4	meine	0.26	gewitter	0.28
5	bin	0.25	flachland	0.26
6	sie	0.25	bis	0.26
7	mir	0.24	alpen	0.24
8	oder	0.24	regen	0.24
9	habe	0.23	weatherwatch	0.24

10	wenn	0.23	hhni	0.24
11	grüsse	0.23	am	0.23
12	mich	0.23	gletscher	0.23
13	ihnen	0.23	niederschlag	0.22
14	eine	0.23	bergen	0.22
Topic 12				
0	film	0.43	ubs	0.51
1	filme	0.35	credit	0.45
2	the	0.34	cs	0.45
3	barbie	0.3	suisse	0.43
4	netflix	0.29	bank	0.4
5	schauspieler	0.29	milliarden	0.32
6	hollywood	0.28	bankenkrise	0.32
7	oscar	0.27	banken	0.31
8	oppenheimer	0.25	übernahme	0.31
9	streik	0.25	grossbank	0.3
10	schauspielerinnen	0.24	snb	0.28
11	kino	0.24	nationalbank	0.28
12	kinos	0.22	franken	0.27
13	of	0.22	finma	0.27
14	studios	0.22	bund	0.25
Topic 13				
0	post	0.49	stau	0.41
1	postfinance	0.4	sbb	0.4
2	weko	0.33	kilometern	0.35
3	spam	0.29	kilometer	0.33
4	henrique	0.28	gotthardbasistunnel	0.32
5	trinkgeld	0.27	länge	0.32
6	gewerkschaft	0.27	tunnel	0.32
7	sgv	0.27	tcs	0.32
8	quickmail	0.26	unfall	0.31
9	pakete	0.26	züge	0.31
10	schneider	0.26	gotthardnordportal	0.3
11	lohn	0.25	zwischen	0.29
12	espresso	0.24	gesperrt	0.29
13	uaw	0.24	richtung	0.29
14	angestellten	0.24	viasuisse	0.29
Topic 14				

0	mitholz	0.45	streik	0.37
1	wef	0.44	gewerkschaft	0.32
2	harald	0.36	bahn	0.3
3	räumung	0.36	polizei	0.3
4	dlr	0.36	streiks	0.29
5	zaun	0.35	aktivisten	0.28
6	räte	0.33	gdl	0.27
7	uli	0.33	demonstration	0.25
8	köhler	0.32	verdi	0.23
9	dr	0.29	warnstreik	0.23
10	ubi	0.29	flughafen	0.22
11	vbs	0.29	am	0.22
12	entscheide	0.29	ausschreitungen	0.21
13	wir	0.28	flüge	0.21
14	planeten	0.28	evg	0.21
Topic 15				
0	behinderungen	0.52	prozent	0.49
1	21	0.51	milliarden	0.37
2	behinderung	0.48	inflation	0.37
3	20	0.43	quartal	0.36
4	usa	0.4	franken	0.36
5	jam	0.39	leitzins	0.36
6	infirmis	0.36	zinsen	0.33
7	26	0.36	snb	0.32
8	behindertensession	0.36	umsatz	0.3
9	eth	0.36	fed	0.3
10	gehenkimberly	0.35	nationalbank	0.29
11	202319	0.35	teuerung	0.28
12	ken	0.35	jahr	0.26
13	kilometern	0.35	lonza	0.26
14	staulänge	0.34	millionen	0.26
Topic 16				
0	gewinner	0.88	charles	0.64
1	gewinnerin	0.81	könig	0.58
2	teilnahmebedingungen	0.76	iii	0.47
3	wohnsitzbestätigung	0.71	krönung	0.46
4	chf	0.7	prinz	0.45
5	anruf	0.68	beckenbauer	0.42

6	mobiltelefone	0.66	berlusconi	0.37
7	teilnahmen	0.64	königin	0.37
8	sponsor	0.61	camilla	0.37
9	ausgeschlossen	0.6	queen	0.34
10	gewinnerinnen	0.6	königshaus	0.31
11	sendung	0.58	ii	0.3
12	gratis	0.57	william	0.29
13	0901	0.57	harry	0.29
14	wettbewerb	0.55	monarchie	0.28
Topic 17				
0	liit	0.44	ich	0.49
1	training	0.43	guten	0.38
2	sportwissenschaftler	0.41	abend	0.35
3	knechtle	0.38	zyklus	0.32
4	snacks	0.38	kann	0.32
5	muskeln	0.37	stuhlgang	0.31
6	gecko	0.36	menopause	0.31
7	aurum	0.35	hitzewallungen	0.31
8	beweglichkeit	0.35	bin	0.31
9	trainiert	0.35	stimme	0.31
10	ems	0.34	hormone	0.31
11	muskel	0.34	habe	0.3
12	sport	0.34	meine	0.3
13	kraft	0.34	radu	0.29
14	hiit	0.34	dr	0.29
Topic 18				
0	rolex	0.9	gewinner	0.65
1	bucherer	0.83	gewinnerin	0.61
2	uhren	0.7	teilnahmebedingungen	0.54
3	swatch	0.69	jackpot	0.51
4	luxusuhren	0.5	wohnsitzbestätigung	0.5
5	akku	0.47	chf	0.5
6	omega	0.46	swisslos	0.47
7	moonswatch	0.4	teilnahmen	0.47
8	marken	0.4	mobiltelefone	0.47
9	bohrschrauber	0.39	preissponsor	0.45
10	grouet	0.37	gewinnerinnen	0.43
11	jorg	0.36	ausgeschlossen	0.42

12	uhrenindustrie	0.35	lotto	0.42
13	greisler	0.35	sendung	0.4
14	smartwatcher	0.35	0901	0.4
Topic 19				
0	jackpot	0.88	rte	1.25
1	lotto	0.82	sommersession	1.13
2	swisslos	0.76	entscheide	1.09
3	gewinn	0.52	liveticker	1.09
4	franken	0.51	wichtigsten	1.04
5	blitz	0.5	eidgenssischen	1.01
6	geknackt	0.46	sondersession	0.94
7	wahrscheinlichkeit	0.44	findet	0.9
8	glückszahl	0.44	statt	0.86
9	gewinne	0.42	frhjahrssession	0.85
10	rappen	0.41	geschfte	0.8
11	zahlenlotto	0.41	geschftslast	0.79
12	weibel	0.41	sondersessionen	0.76
13	ausbezahlt	0.41	überblick	0.74
14	geburtstagsdatum	0.4	agenturen	0.72

Table 18: This compares all topic modeling results from the SRF corpus between both encoder models.

Numbering	**Sentence-BERT**		**SentenceSwissBERT**	
Topic 0				
0	été	0.21	film	0.22
1	ont	0.2	son	0.21
2	police	0.18	sa	0.19
3	dans	0.18	et	0.19
4	personnes	0.18	qui	0.19
5	un	0.18	un	0.19
6	sur	0.18	avec	0.19
7	incendie	0.18	dans	0.18
8	une	0.17	ce	0.18
9	incendies	0.17	voir	0.18
10	sont	0.17	une	0.18
11	des	0.17	scène	0.18
12	les	0.17	est	0.18
13	après	0.17	il	0.18

14	pompiers	0.17	ses	0.18
Topic 1				
0	film	0.23	guerre	0.23
1	son	0.21	pays	0.22
2	scène	0.21	gaza	0.21
3	festival	0.2	hamas	0.2
4	album	0.2	israël	0.2
5	théâtre	0.2	l'armée	0.2
6	sa	0.19	russie	0.19
7	et	0.19	l'ukraine	0.19
8	avec	0.19	l'onu	0.18
9	the	0.19	que	0.18
10	musique	0.19	russe	0.18
11	voir	0.18	au	0.18
12	roman	0.18	chine	0.18
13	un	0.18	ont	0.18
14	qui	0.18	dans	0.18
Topic 2				
0	crédit	0.29	ont	0.23
1	suisse	0.27	pompiers	0.21
2	banque	0.26	dans	0.2
3	ubs	0.24	personnes	0.2
4	banques	0.2	incendies	0.2
5	taux	0.2	police	0.19
6	bns	0.19	contenu	0.19
7	milliards	0.19	selon	0.19
8	bancaire	0.18	feu	0.19
9	que	0.18	autorités	0.19
10	pas	0.18	sont	0.18
11	sur	0.18	plus	0.18
12	en	0.18	des	0.18
13	des	0.18	morts	0.18
14	agriculteurs	0.17	degrés	0.18
Topic 3				
0	ukraine	0.29	on	0.19
1	russie	0.26	plus	0.19
2	russe	0.26	sont	0.19
3	russes	0.22	ou	0.19

4	guerre	0.22	les	0.19
5	poutine	0.21	est	0.18
6	migrants	0.2	dans	0.18
7	pays	0.2	pour	0.18
8	vladimir	0.2	en	0.18
9	asile	0.19	pas	0.18
10	turquie	0.19	une	0.18
11	moscou	0.19	que	0.18
12	président	0.18	des	0.18
13	erdogan	0.18	suisse	0.18
14	ukrainien	0.18	et	0.18
Topic 4				
0	conseil	0.24	projet	0.22
1	canton	0.22	primes	0.21
2	parti	0.19	conseil	0.21
3	plr	0.19	francs	0.21
4	udc	0.19	fédéral	0.21
5	fédéral	0.19	coûts	0.2
6	état	0.19	pour	0.19
7	réforme	0.19	loi	0.19
8	au	0.18	les	0.19
9	du	0.18	canton	0.19
10	vert	0.18	cantons	0.19
11	pour	0.18	l'initiative	0.19
12	il	0.18	maladie	0.18
13	pas	0.18	par	0.18
14	conseiller	0.18	des	0.18
Topic 5				
0	israël	0.29	trump	0.33
1	gaza	0.27	donald	0.32
2	hamas	0.26	président	0.25
3	armée	0.22	présidentielle	0.25
4	l'armée	0.21	présidentielles	0.22
5	palestiniens	0.2	présidente	0.21
6	l'etat	0.2	élection	0.2
7	palestinien	0.19	joe	0.2
8	la	0.19	kamala	0.19
9	militaires	0.18	les	0.19

10	palestinienne	0.18	élections	0.19
11	est	0.18	biden	0.19
12	sur	0.18	démocrates	0.19
13	l'état	0.18	les	0.18
14	depuis	0.18	et	0.18
Topic 6				
0	eau	0.22	grève	0.3
1	loup	0.2	d'asile	0.24
2	loups	0.19	agriculteurs	0.23
3	plus	0.19	syndicat	0.22
4	électricité	0.18	personnes	0.2
5	les	0.18	cicr	0.19
6	degrés	0.18	ont	0.19
7	des	0.18	syndicats	0.19
8	sont	0.18	des	0.19
9	en	0.18	les	0.19
10	climatique	0.18	mobilisation	0.19
11	est	0.18	pour	0.18
12	dans	0.18	du	0.18
13	pour	0.18	en	0.18
14	on	0.18	france	0.18
Topic 7				
0	finale	0.32	conseil	0.33
1	00	0.32	plr	0.33
2	match	0.29	parti	0.31
3	débute	0.27	ps	0.29
4	league	0.27	l'udc	0.29
5	coupe	0.24	élections	0.28
6	fc	0.24	états	0.28
7	3e	0.23	fédérales	0.27
8	espagne	0.23	national	0.26
9	2e	0.22	candidats	0.26
10	servette	0.22	berset	0.26
11	pedro	0.22	sièges	0.26
12	matches	0.22	sige	0.26
13	football	0.21	partis	0.26
14	djokovic	0.21	fédéral	0.26
Topic 8				

0	santé	0.26	tribunal	0.26
1	maladie	0.26	procès	0.24
2	primes	0.25	tariq	0.23
3	tabac	0.24	ramadan	0.23
4	médicaments	0.23	avait	0.23
5	virus	0.23	prison	0.23
6	assurance	0.22	cour	0.22
7	covid	0.21	son	0.21
8	cannabis	0.2	justice	0.2
9	coûts	0.2	ans	0.2
10	vaccin	0.2	depardieu	0.2
11	cancer	0.19	faits	0.2
12	patients	0.19	bolsonaro	0.19
13	patient	0.19	ministre	0.19
14	cigarettes	0.18	sa	0.19
Topic 9				
0	église	0.28	nasa	0.24
1	abus	0.27	artificielle	0.23
2	femmes	0.26	spatiale	0.23
3	catholique	0.26	lune	0.22
4	sexuels	0.25	scientifiques	0.22
5	pape	0.25	ia	0.21
6	tariq	0.22	mission	0.21
7	ramadan	0.22	lintelligence	0.21
8	vatican	0.21	spatial	0.2
9	avortement	0.21	chatgpt	0.2
10	elle	0.19	fusée	0.2
11	abbaye	0.19	est	0.19
12	françois	0.19	galaxies	0.19
13	une	0.18	lunivers	0.19
14	il	0.18	plus	0.19
Topic 10				
0	ia	0.34	taux	0.29
1	intelligence	0.3	hausse	0.29
2	artificielle	0.3	l'inflation	0.27
3	chatgpt	0.25	prix	0.27
4	données	0.25	millions	0.26
5	tiktok	0.24	croissance	0.25

6	utilisateurs	0.23	2022	0.24
7	musk	0.23	budget	0.22
8	elon	0.23	francs	0.22
9	google	0.21	pandémie	0.21
10	meta	0.2	nuites	0.21
11	twitter	0.2	milliards	0.21
12	technologie	0.19	suisse	0.21
13	contenus	0.19	chômage	0.2
14	ou	0.19	l'année	0.2
Topic 11				
0	trump	0.41	débute	0.41
1	donald	0.38	finale	0.41
2	biden	0.36	match	0.41
3	joe	0.34	league	0.38
4	président	0.32	atp	0.31
5	présidentielle	0.28	djokovic	0.29
6	républicain	0.26	fc	0.29
7	américain	0.24	matches	0.29
8	bolsonaro	0.23	3e	0.27
9	ancien	0.23	wta	0.27
10	blanche	0.22	chelem	0.27
11	jair	0.22	bienne	0.26
12	démocrate	0.22	score	0.26
13	unis	0.22	novak	0.26
14	républicains	0.21	monde	0.32
Topic 12				
0	chine	0.4	crédit	0.49
1	taïwan	0.35	banque	0.38
2	pékin	0.34	ubs	0.35
3	chinois	0.32	suisse	0.33
4	corée	0.28	rachat	0.32
5	japon	0.25	finma	0.3
6	chinoise	0.24	banques	0.29
7	xi	0.24	bancaire	0.29
8	jinping	0.24	cep	0.27
9	nord	0.23	marchés	0.26
10	sud	0.22	bns	0.25
11	île	0.21	milliards	0.24

12	titanic	0.21	actionnaires	0.24
13	submersible	0.2	postes	0.24
14	unis	0.2	faillite	0.23
Topic 13				
0	vous	0.31	tiktok	0.36
1	rts	0.29	données	0.32
2	actualité	0.28	twitter	0.32
3	cicr	0.27	xplain	0.28
4	rtsinfo	0.25	musk	0.28
5	min	0.25	elon	0.26
6	extraits	0.24	cyberattaque	0.26
7	nos	0.23	plateforme	0.24
8	good	0.23	cocaïne	0.24
9	minute	0.23	fedpol	0.23
10	mailchimp	0.22	utilisateurs	0.22
11	suisse	0.22	réseau	0.22
12	interviews	0.22	pirates	0.22
13	vu	0.22	informatiques	0.21
14	7h	0.22	microsoft	0.21
Topic 14				
0	lune	0.4	odermatt	0.37
1	nasa	0.39	coupe	0.37
2	spatiale	0.36	podium	0.33
3	fusée	0.34	superg	0.33
4	esa	0.32	2e	0.33
5	mission	0.32	monde	0.32
6	galaxies	0.3	étape	0.32
7	spatial	0.3	discipline	0.31
8	étoiles	0.3	gutbehrami	0.31
9	lunaire	0.29	3e	0.31
10	galaxie	0.28	descente	0.31
11	spacex	0.28	marco	0.3
12	étoile	0.28	médaille	0.3
13	sonde	0.27	slalom	0.3
14	euclid	0.27	4e	0.3
Topic 15				
0	scénaristes	0.42	festival	0.42
1	studios	0.35	édition	0.37

2	poste	0.34	concerts	0.3
3	grève	0.34	fête	0.28
4	acteurs	0.33	organisateurs	0.28
5	hollywood	0.33	musique	0.28
6	migros	0.31	artistes	0.27
7	friday	0.27	jazz	0.26
8	redevance	0.27	manifestation	0.26
9	postes	0.26	programmation	0.22
10	syndicat	0.26	soirées	0.21
11	wga	0.26	musicales	0.21
12	aftra	0.24	du	0.21
13	concession	0.24	d'artifice	0.2
14	sag	0.24	année	0.2
Topic 16				
0	roi	0.58	tunnel	0.57
1	charles	0.54	gothard	0.53
2	iii	0.52	cff	0.48
3	reine	0.47	trafic	0.46
4	ii	0.46	trains	0.42
5	elizabeth	0.41	ferroviaire	0.39
6	couronnement	0.4	marchandises	0.38
7	prince	0.39	l'autoroute	0.37
8	visite	0.38	déraillement	0.34
9	camilla	0.34	ligne	0.33
10	royale	0.33	panne	0.32
11	britannique	0.33	voyageurs	0.32
12	monarque	0.31	bouchon	0.3
13	buckingham	0.31	perturbé	0.3
14	harry	0.3	travaux	0.3
Topic 17				
0	noël	0.63	extraits	0.71
1	restaurant	0.34	vu	0.64
2	plats	0.34	7h	0.64
3	thefork	0.33	interviews	0.64
4	père	0.33	actualité	0.61
5	plat	0.31	vous	0.6
6	pourboire	0.3	rtsinfo	0.59
7	restaurants	0.29	débat	0.59

8	palourdes	0.27	démissions	0.58
9	pourboires	0.27	déclarations	0.55
10	crabes	0.27	analyses	0.53
11	freddo	0.25	vidéo	0.53
12	bô	0.25	forum	0.52
13	restaurateurs	0.25	revient	0.52
14	michelin	0.25	veille	0.51
Topic 18				
0	abeilles	1.05	roi	0.63
1	miel	0.61	iii	0.6
2	ruches	0.58	charles	0.59
3	colonies	0.55	reine	0.53
4	apiculteurs	0.51	couronnement	0.48
5	sucre	0.41	ii	0.46
6	apiculture	0.4	visite	0.4
7	abeille	0.38	elizabeth	0.39
8	mellifères	0.38	camilla	0.38
9	biodiversité	0.38	monarque	0.37
10	roundup	0.37	westminster	0.34
11	pollinisateurs	0.36	britannique	0.34
12	disparition	0.36	d'elizabeth	0.33
13	sauvages	0.35	buckingham	0.33
14	miels	0.34	royaume-uni	0.33
Topic 19				
0	archéologues	0.49	michelin	0.69
1	dalle	0.47	guide	0.53
2	pierres	0.46	cuisinier	0.52
3	bélec	0.45	restaurant	0.48
4	curie	0.45	carcenat	0.43
5	desert	0.43	étoiles	0.43
6	tumulus	0.42	vins	0.43
7	jordanie	0.4	gastronomique	0.41
8	squelettes	0.4	rougemont	0.4
9	barge	0.4	gaultmillau	0.39
10	pailler	0.39	chevrier	0.38
11	kite	0.39	benoît	0.38
12	pompéi	0.38	étoile	0.38
13	gravés	0.38	gastronomie	0.38

14	fosses	0.38	robert	0.38

Table 18: This compares all topic modeling results from the RTS corpus between both encoder models.

Numbering	**Sentence-BERT**		**SentenceSwissBERT**	
		Topic 0		
0	finale	0.21	set	0.25
1	al	0.20	finale	0.24
2	con	0.20	al	0.24
3	gol	0.19	gol	0.24
4	dopo	0.19	punti	0.24
5	ha	0.19	hanno	0.23
6	primo	0.19	primo	0.23
7	partita	0.19	grazie	0.23
8	punti	0.19	con	0.23
9	nella	0.19	partita	0.22
10	prima	0.18	match	0.22
11	in	0.18	dopo	0.22
12	squadra	0.18	casa	0.22
13	set	0.18	contro	0.22
14	contro	0.18	vantaggio	0.21
		Topic 1		
0	credit	0.23	lugano	0.27
1	ubs	0.22	squadra	0.25
2	suisse	0.22	stagione	0.25
3	banca	0.22	partite	0.24
4	svizzera	0.19	crocitorti	0.24
5	le	0.18	partita	0.23
6	della	0.18	campionato	0.22
7	dei	0.18	gol	0.21
8	franchi	0.18	league	0.21
9	che	0.18	ma	0.21
10	delle	0.18	club	0.21
11	dell	0.18	con	0.2
12	un	0.18	35	0.2
13	una	0.18	contro	0.2
14	anche	0.18	non	0.2

Topic 2				
0	polizia	0.24	gara	0.32
1	persone	0.22	podio	0.29
2	sono	0.2	tappa	0.27
3	incendio	0.2	sprint	0.24
4	fiamme	0.2	nella	0.24
5	un	0.19	posto	0.24
6	una	0.19	tour	0.23
7	pompieri	0.19	mondo	0.23
8	le	0.19	100m	0.23
9	stato	0.19	giro	0.22
10	si	0.18	chiuso	0.22
11	uomo	0.18	prova	0.22
12	cantonale	0.18	traguardo	0.22
13	incidente	0.18	ha	0.21
14	da	0.18	gp	0.21
Topic 3				
0	ucraina	0.30	russia	0.23
1	russia	0.27	mosca	0.21
2	kiev	0.26	presidente	0.21
3	mosca	0.24	guerra	0.2
4	russo	0.23	ha	0.2
5	putin	0.23	gaza	0.2
6	ucraino	0.22	russo	0.2
7	presidente	0.22	putin	0.2
8	guerra	0.22	israele	0.19
9	russi	0.21	ucraina	0.19
10	cina	0.21	militare	0.19
11	zelensky	0.21	forze	0.19
12	russa	0.2	stati	0.19
13	wagner	0.2	della	0.19
14	ha	0.2		
Topic 4				
0	galleria	0.32	persone	0.22
1	traffico	0.31	sono	0.22
2	gottardo	0.31	metri	0.22
3	treni	0.29	le	0.2
4	ffs	0.29	zona	0.2

5	san	0.27	gradi	0.2
6	passeggeri	0.26	fiamme	0.2
7	luna	0.25	migranti	0.2
8	spaziale	0.25	si	0.19
9	merci	0.24	precipitazioni	0.19
10	tunnel	0.24	delle	0.19
11	volo	0.23	pompieri	0.19
12	chilometri	0.23	vento	0.19
13	direzione	0.23	costiera	0.19
14	portale	0.23	temperature	0.19
Topic 5				
0	partito	0.29	credit	0.3
1	elezioni	0.27	suisse	0.29
2	verdi	0.27	ubs	0.29
3	consiglio	0.25	banca	0.26
4	udc	0.24	consiglio	0.21
5	voti	0.24	svizzera	0.21
6	plr	0.24	banche	0.19
7	governo	0.23	federale	0.19
8	riforma	0.23	dei	0.19
9	ps	0.23	delle	0.19
10	voto	0.23	le	0.19
11	candidati	0.23	che	0.19
12	seggi	0.22	non	0.19
13	sinistra	0.22	una	0.19
14	federali	0.21	franchi	0.19
Topic 6				
0	animali	0.3	polizia	0.33
1	gradi	0.29	cantonale	0.26
2	lupi	0.27	luomo	0.23
3	temperature	0.26	ferite	0.22
4	lupo	0.26	un	0.22
5	animale	0.26	stato	0.21
6	specie	0.25	una	0.21
7	caldo	0.23	donna	0.21
8	caccia	0.22	era	0.2
9	branchi	0.22	anni	0.2
10	esemplari	0.22	due	0.2

11	acqua	0.21	cocaina	0.2
12	abbattimento	0.2	conducente	0.2
13	orso	0.2	della	0.2
14	temperatura	0.19	stata	0.19
Topic 7				
0	gaza	0.39	prodotti	0.21
1	israele	0.36	territorio	0.21
2	hamas	0.35	animali	0.2
3	palestinesi	0.31	produzione	0.2
4	israeliano	0.3	come	0.19
5	striscia	0.3	luogo	0.19
6	palestinese	0.29	anche	0.19
7	netanyahu	0.26	che	0.19
8	israeliani	0.26	le	0.19
9	esercito	0.26	da	0.19
10	ostaggi	0.25	valle	0.19
11	niger	0.23	non	0.19
12	cessate	0.22	qualit	0.19
13	attacchi	0.22	un	0.18
14	cisgiordania	0.22	per	0.18
Topic 8				
0	trump	0.47	film	0.36
1	biden	0.36	festival	0.31
2	donald	0.34	anni	0.25
3	presidente	0.31	cinema	0.23
4	joe	0.28	premio	0.23
5	uniti	0.27	regista	0.22
6	bianca	0.26	aveva	0.21
7	repubblicano	0.26	locarno	0.21
8	primarie	0.24	era	0.2
9	repubblicani	0.24	lattore	0.2
10	corte	0.23	edizione	0.2
11	procuratore	0.22	della	0.2
12	congresso	0.22	cantante	0.2
13	stati	0.22	sua	0.2
14	di	0.22	grande	0.19
Topic 9				
0	berlusconi	0.32	trump	0.42

1	cocaina	0.3	presidente	0.31
2	silvio	0.25	biden	0.3
3	covid	0.24	donald	0.3
4	sanità	0.23	elezioni	0.28
5	droga	0.23	lex	0.26
6	pazienti	0.22	partito	0.25
7	medici	0.22	presidenziali	0.24
8	sigarette	0.22	joe	0.24
9	vaccino	0.21	repubblicano	0.23
10	italia	0.21	elettorale	0.23
11	virus	0.21	primarie	0.23
12	farmaci	0.2	destra	0.23
13	malattie	0.2	voto	0.23
14	vaccini	0.2	repubblicani	0.23
Topic 10				
0	film	0.44	tasso	0.29
1	festival	0.31	miliardi	0.29
2	cinema	0.3	milioni	0.29
3	attore	0.3	prezzi	0.28
4	regista	0.29	tassi	0.28
5	carlo	0.28	franchi	0.28
6	cantante	0.26	crescita	0.26
7	anni	0.26	rispetto	0.26
8	premio	0.25	aumento	0.25
9	re	0.24	2022	0.24
10	sua	0.23	2023	0.24
11	aveva	0.23	linflazione	0.23
12	miglior	0.23	calo	0.23
13	iii	0.23	trimestre	0.23
14	picasso	0.22	dei	0.22
Topic 11				
0	energia	0.39	protesta	0.27
1	co2	0.33	sindacati	0.27
2	impianti	0.3	sciopero	0.26
3	elettrica	0.29	polizia	0.26
4	elettriche	0.27	manifestanti	0.25
5	elettricità	0.27	agricoltori	0.25
6	impianto	0.27	manifestazione	0.24

7	gas	0.26	parigi	0.24
8	rinnovabili	0.26	mobilitazione	0.23
9	auto	0.26	scontri	0.22
10	solare	0.25	persone	0.22
11	emissioni	0.25	contadini	0.22
12	approvvigionamento	0.24	trattori	0.21
13	produzione	0.24	dellordine	0.21
14	energetica	0.24	disordini	0.21
Topic 12				
0	intervista	1.2	progetto	0.28
1	behrami	0.91	progetti	0.25
2	manche	0.91	impianti	0.24
3	gut	0.9	costruzione	0.22
4	lara	0.87	energia	0.22
5	odermatt	0.74	traffico	0.22
6	analizza	0.68	solare	0.22
7	luca	0.67	elettrica	0.21
8	gigante	0.65	realizzazione	0.21
9	marco	0.65	impianto	0.21
10	daniele	0.61	solari	0.2
11	grassi	0.6	per	0.2
12	dichiarazioni	0.6	lavori	0.2
13	courchevel	0.59	rinnovabili	0.2
14	premiazione	0.59	emissioni	0.2
Topic 13				
0	intelligenza	0.35	partito	0.36
1	artificiale	0.33	verdi	0.34
2	utenti	0.31	candidati	0.31
3	dati	0.31	elezioni	0.31
4	musk	0.3	lista	0.31
5	hacker	0.27	consiglio	0.31
6	chatgpt	0.26	ps	0.29
7	meta	0.25	liste	0.29
8	facebook	0.25	partiti	0.29
9	social	0.25	plr	0.28
10	informatici	0.24	federali	0.28
11	piattaforme	0.23	seggi	0.28
12	ia	0.23	seggio	0.28

13	software	0.23	federale	0.25
14	elon	0.23	liberali	0.25
Topic 14				
0	papa	0.45	fifa	0.29
1	abusi	0.4	riga	0.27
2	francesco	0.39	zelanda	0.25
3	chiesa	0.39	mondiali	0.24
4	diocesi	0.37	fischer	0.24
5	vescovo	0.35	calcio	0.23
6	santa	0.31	mondiale	0.23
7	sessuali	0.3	forrer	0.23
8	apostolico	0.29	svezia	0.23
9	vaticano	0.29	remo	0.22
10	cattolica	0.29	grings	0.22
11	pontefice	0.28	svizzera	0.22
12	vescovi	0.26	giochi	0.22
13	raemy	0.25	giocatori	0.21
14	pace	0.24	inka	0.21
Topic 15				
0	rubiales	1.02	galleria	0.45
1	hermoso	0.94	gottardo	0.43
2	bacio	0.85	traffico	0.41
3	spagnola	0.75	treni	0.4
4	calciatrice	0.64	ffs	0.39
5	luis	0.63	san	0.38
6	jennifer	0.6	merci	0.34
7	rfef	0.59	portale	0.32
8	premiazione	0.57	direzione	0.32
9	giocatrice	0.56	tunnel	0.32
10	calcio	0.56	code	0.31
11	federcalcio	0.55	passeggeri	0.31
12	procura	0.53	airolo	0.3
13	federazione	0.53	gschenen	0.3
14	sessuale	0.5	viaggiatori	0.29
Topic 16				
0	pride	0.74	papa	0.37
1	lgbtiq	0.52	berlusconi	0.35
2	gay	0.49	francesco	0.32

3	corteo	0.45	abusi	0.3
4	comunità	0.43	chiesa	0.29
5	istanbul	0.43	diocesi	0.28
6	eurogames	0.43	vescovo	0.27
7	manifestazione	0.4	silvio	0.27
8	nrk	0.38	carlo	0.26
9	queer	0.38	santa	0.25
10	oslo	0.38	ricoverato	0.24
11	manifestanti	0.38	vaticano	0.23
12	arcobaleno	0.37	apostolico	0.22
13	persone	0.36	italia	0.22
14	lesbiche	0.35	re	0.22
Topic 17				
0	servizio	1.71	ruag	0.38
1	u13	1.01	carri	0.32
2	finestrella	0.98	armati	0.31
3	sportsera	0.89	leopard	0.3
4	neonato	0.87	federale	0.29
5	municipal	0.84	documenti	0.26
6	scopri	0.82	hacker	0.25
7	brisbane	0.79	rheinmetall	0.23
8	six	0.79	confederazione	0.23
9	sulla	0.78	dati	0.22
10	info	0.78	informatici	0.22
11	approfondimento	0.73	assange	0.22
12	nasconde	0.72	difesa	0.21
13	torneo	0.72	informazioni	0.21
14	faido	0.72	bellico	0.21
Topic 18				
0	noè	1.71	spaziale	0.58
1	ponti	1.68	luna	0.52
2	delfino	1.57	missione	0.48
3	200m	1.42	nasa	0.45
4	premiazione	1.11	lunare	0.45
5	corta	1.06	sonda	0.4
6	vasca	1.02	razzo	0.4
7	otopeni	1.01	lancio	0.37
8	50m	1.0	lander	0.36

9	vinta	0.91	satellite	0.35
10	rana	0.91	spacex	0.34
11	100m	0.9	astronauti	0.34
12	dorso	0.89	decollo	0.33
13	immagini	0.87	orbita	0.3
14	finale	0.86	starship	0.29
Topic 19				
0	docenti	0.65	lintervista	0.77
1	sedi	0.6	19h05	0.65
2	allievi	0.57	parler	0.58
3	sperimentazione	0.57	supplementari	0.57
4	acquarossa	0.54	19h10	0.56
5	scuola	0.54	puntata	0.51
6	scolastico	0.52	rete	0.51
7	scuole	0.52	consueto	0.47
8	matematica	0.5	tempi	0.46
9	caslano	0.48	calcio	0.45
10	docenza	0.44	uno	0.43
11	classi	0.44	appuntamento	0.42
12	moutier	0.43	su	0.42
13	coinvolgerà	0.42	dedicata	0.4
14	insegnamento	0.4	trasmissione	0.38

Table 18: This compares all topic modeling results from the RSI corpus between both encoder models.

Numbering	**Sentence-BERT**		**SentenceSwissBERT**	
Topic 0				
0	cussegl	0.2	vischnanca	0.3
1	per	0.19	francs	0.29
2	vischnanca	0.19	project	0.29
3	ch	0.18	communala	0.27
4	che	0.18	approv	0.25
5	dal	0.18	milliuns	0.23
6	onn	0.18	radunanza	0.23
7	las	0.18	ovras	0.22
8	ina	0.18	credit	0.22
9	francs	0.18	construcziun	0.22
10	grischun	0.18	bogn	0.21

11	cun	0.17	abitaziuns	0.21
12	ils	0.17	communal	0.21
13	ed	0.17	solara	0.21
14	in	0.17	per	0.2
Topic 1				
0	cursa	0.38	tabella	0.41
1	skis	0.29	hockey	0.41
2	cuppa	0.27	hcd	0.4
3	cursas	0.27	gieu	0.39
4	mundiala	0.27	liga	0.39
5	podest	0.26	gol	0.36
6	secundas	0.25	cunter	0.35
7	campiunadis	0.24	terz	0.34
8	gudagnà	0.24	tavau	0.33
9	mountainbike	0.24	gols	0.33
10	mundials	0.24	club	0.32
11	sport	0.24	lequipa	0.32
12	schurter	0.23	ballape	0.31
13	swiss	0.23	minuta	0.31
14	partenza	0.22	federaziun	0.3
2	**Topic 2**			
0	hcd	0.41	rtr	0.36
1	tabella	0.4	quartier	0.32
2	gieu	0.39	schlifras	0.28
3	liga	0.38	58	0.24
4	hockey	0.38	rain	0.24
5	gol	0.34	lai	0.23
6	ehc	0.33	deflorin	0.23
7	equipa	0.33	67	0.23
8	cunter	0.33	flavio	0.22
9	terz	0.32	onns	0.22
10	gols	0.31	clois	0.21
11	tavau	0.31	fotografias	0.21
12	ballape	0.3	teater	0.2
13	minuta	0.29	fotografia	0.2
14	federaziun	0.29	hardegger	0.19
Topic 3				
0	ucraina	0.35	cursa	0.42

1	russia	0.32	schurter	0.32
2	russa	0.26	plaz	0.3
3	ucranais	0.25	secundas	0.3
4	president	0.24	epic	0.3
5	russ	0.23	podest	0.3
6	selenski	0.23	etappa	0.3
7	guerra	0.23	mundiala	0.29
8	armas	0.23	cuppa	0.29
9	ucranaisa	0.23	gudagn	0.28
10	putin	0.23	cursas	0.28
11	pajais	0.22	nino	0.28
12	moscau	0.22	duo	0.27
13	haja	0.21	cape	0.25
14	russas	0.21	barandun	0.25
Topic 4				
0	traffic	0.48	lucraina	0.26
1	gottard	0.41	gaza	0.26
2	colonnas	0.39	strivla	0.26
3	a13	0.37	lisrael	0.24
4	tunnel	0.37	russia	0.24
5	trens	0.36	hamas	0.24
6	vias	0.35	larmada	0.22
7	colonna	0.34	guerra	0.22
8	autostrada	0.34	israeliana	0.21
9	pasca	0.29	sajan	0.2
10	sid	0.29	attatgas	0.2
11	astra	0.28	stadis	0.2
12	retica	0.28	tenor	0.2
13	portal	0.28	lonu	0.2
14	viafier	0.28	ha	0.2
Topic 5				
0	polizia	0.52	scola	0.26
1	incendis	0.43	surselva	0.24
2	pumpiers	0.42	suprastanza	0.24
3	incendi	0.38	sanadad	0.24
4	fieu	0.37	communal	0.23
5	chantunala	0.37	communala	0.22
6	guaud	0.34	rumantscha	0.22

7	accident	0.31	radio	0.22
8	stizzar	0.31	lia	0.21
9	rut	0.29	nov	0.21
10	bitsch	0.29	el	0.21
11	acziun	0.29	vischnancas	0.21
12	auto	0.3	sco	0.21
13	um	0.28	persunal	0.2
14	derasà	0.26	eleg	0.2
Topic 6				
0	quartier	0.47	polizia	0.38
1	rtr	0.36	pumpiers	0.32
2	schlifras	0.35	plievgia	0.31
3	rain	0.34	chantunala	0.3
4	deflorin	0.33	naiv	0.3
5	flavio	0.33	fieu	0.29
6	58	0.31	incendis	0.27
7	lai	0.3	grads	0.26
8	fotografias	0.28	lavinas	0.26
9	clois	0.27	lincendi	0.25
10	fotografia	0.27	privel	0.25
11	chasas	0.27	guaud	0.25
12	rischatsch	0.27	meteo	0.25
13	quartiers	0.26	hai	0.24
14	139	0.25	stizzar	0.23
Topic 7				
0	gaza	0.37	lonn	0.29
1	strivla	0.37	pretschs	0.29
2	israel	0.36	tschains	0.27
3	hamas	0.35	cumparegli	0.27
4	israeliana	0.3	premias	0.26
5	agid	0.27	pernottaziuns	0.26
6	ostagis	0.26	pli	0.26
7	palestinais	0.26	dapli	0.26
8	armada	0.25	2022	0.24
9	onu	0.25	malsauns	0.24
10	israelian	0.25	milliuns	0.24
11	pausa	0.24	svizra	0.24
12	netanjahu	0.23	francs	0.24

13	persunas	0.23	custs	0.24
14	rauba	0.23	persunas	0.24
Topic 8				
0	catolica	0.64	lufs	0.45
1	baselgia	0.61	chatscha	0.41
2	abus	0.55	triep	0.4
3	papa	0.48	luf	0.38
4	sexual	0.48	trieps	0.36
5	uvestgs	0.41	sajettar	0.31
6	puntraschigna	0.38	animals	0.31
7	uvestg	0.37	pestga	0.3
8	francestg	0.36	peschs	0.29
9	cas	0.34	luffizi	0.27
10	plaiv	0.34	sajettads	0.26
11	bonnemain	0.32	niz	0.26
12	joseph	0.31	chantun	0.26
13	vatican	0.31	regulaziun	0.25
14	studi	0.29	grischun	0.25
Topic 9				
0	posta	0.59	traffic	0.54
1	pachets	0.5	gottard	0.41
2	postfinance	0.49	vias	0.4
3	taxa	0.49	colonnas	0.4
4	brevs	0.43	lautostrada	0.38
5	distribuziun	0.35	tunnel	0.37
6	quickmail	0.35	colonna	0.37
7	srg	0.34	pasca	0.36
8	ponn	0.33	trens	0.35
9	suvs	0.33	via	0.32
10	finma	0.32	a13	0.31
11	francs	0.31	retica	0.31
12	fabrica	0.3	portal	0.31
13	luna	0.29	bernardino	0.3
14	duain	0.29	viafier	0.29
Topic 10				
0	trump	0.78	cussegl	0.41
1	donald	0.59	banca	0.39
2	berset	0.55	suisse	0.35

3	alain	0.52	chantuns	0.34
4	preelecziuns	0.41	credit	0.34
5	candidatura	0.4	federal	0.33
6	president	0.39	cs	0.32
7	anteriur	0.38	lubs	0.32
8	capitol	0.38	ubs	0.3
9	haley	0.38	naziunal	0.29
10	accusaziun	0.37	milliardas	0.29
11	scalise	0.36	francs	0.28
12	republicans	0.36	surpigliada	0.27
13	sez	0.35	bancas	0.27
14	elecziuns	0.35	liniziativa	0.27
Topic 11				
0	weko	0.54	partida	0.41
1	administrativ	0.52	ps	0.33
2	tribunal	0.51	cussegl	0.32
3	recurs	0.49	pps	0.32
4	foffa	0.48	elecziuns	0.31
5	conrad	0.45	pld	0.3
6	lazzarini	0.42	sez	0.3
7	luf	0.42	cusseglier	0.29
8	lufs	0.42	pult	0.29
9	federal	0.41	vuschs	0.28
10	trieps	0.41	clima	0.28
11	ordinaziun	0.39	sezs	0.28
12	bafu	0.39	naziunal	0.27
13	protecziun	0.39	glistas	0.27
14	gruppa	0.38	jon	0.26
Topic 12				
0	datas	0.66	brinzauls	0.46
1	cyber	0.53	vitg	0.41
2	hackers	0.49	vischnanca	0.4
3	darknet	0.48	bova	0.37
4	sensiblas	0.45	crappa	0.36
5	key	0.45	linsla	0.33
6	it	0.43	privel	0.33
7	xplain	0.43	alvra	0.32
8	fedpol	0.43	fasa	0.32

9	adressas	0.42	moviment	0.31
10	enguladas	0.42	monn	0.31
11	paginas	0.39	crudada	0.29
12	ddos	0.39	levacuaziun	0.29
13	hacker	0.37	casti	0.28
14	logger	0.37	meters	0.28
Topic 13				
0	muments	0.56	wef	0.39
1	brügger	0.51	stizun	0.36
2	58	0.5	vestgadira	0.32
3	44	0.47	container	0.31
4	38	0.46	butia	0.27
5	37	0.46	second	0.26
6	annalea	0.46	hand	0.26
7	kienz	0.45	tavau	0.25
8	mais	0.43	dunnas	0.24
9	ramosch	0.42	caritas	0.23
10	maset	0.41	theo	0.23
11	maria	0.41	demonstraziun	0.23
12	bostg	0.39	era	0.23
13	bels	0.39	chauma	0.23
14	soliva	0.39	di	0.22
Topic 14				
0	drogas	0.69	baselgia	0.41
1	consum	0.55	papa	0.39
2	local	0.45	charles	0.37
3	cannabis	0.44	dabus	0.34
4	cocain	0.41	haas	0.33
5	consumaziun	0.41	el	0.33
6	violenza	0.4	catolica	0.33
7	cuira	0.37	plaiv	0.3
8	scena	0.37	sexual	0.3
9	psichica	0.36	premi	0.3
10	sägenstrasse	0.34	luvestg	0.29
11	prevenziun	0.34	uvestg	0.29
12	dependentas	0.33	wolfgang	0.27
13	degiacomi	0.33	retg	0.27
14	dependenza	0.33	spirituals	0.27

Topic 16				
0	rossa	0.5	accidents	0.47
1	buseno	0.47	bus	0.47
2	sbuvaditschs	0.46	velos	0.43
3	schwanden	0.43	vignetta	0.42
4	plievgia	0.43	ebikes	0.38
5	crappa	0.42	electrics	0.37
6	albertini	0.41	velo	0.37
7	motiv	0.4	daccidents	0.36
8	serrada	0.37	purschida	0.36
9	gianfranco	0.36	battaria	0.33
10	via	0.36	surselva	0.3
11	angehrn	0.36	autos	0.3
12	fräsy	0.36	traffic	0.3
13	crudada	0.35	venda	0.29
14	maraton	0.34	lauto	0.29
Topic 17				
0	etappa	0.74	trump	0.7
1	cape	0.72	donald	0.55
2	frischknecht	0.67	preelecziuns	0.43
3	duo	0.67	erdogan	0.43
4	epic	0.65	kilicdaroglu	0.41
5	beeli	0.64	capitol	0.4
6	barandun	0.62	scalise	0.4
7	schurter	0.62	republicans	0.39
8	andrin	0.58	vuschs	0.39
9	anderes	0.54	president	0.37
10	nino	0.52	partida	0.36
11	fadri	0.49	lanteriur	0.35
12	sandro	0.49	lelecziun	0.35
13	baum	0.49	democrats	0.35
14	prolog	0.49	elecziuns	0.35
Topic 18				
0	premias	0.93	drogas	0.69
1	malsauns	0.67	medicaments	0.59
2	cassas	0.62	consum	0.52
3	cassa	0.53	local	0.51
4	reducziun	0.5	cocain	0.43

5	franschisa	0.49	consumaziun	0.42
6	priminfo	0.44	scena	0.41
7	assicuranza	0.42	dependentas	0.37
8	spargnar	0.39	substanzas	0.36
9	placi	0.38	consumar	0.35
10	custs	0.37	cuira	0.34
11	degonda	0.36	sgenstrasse	0.34
12	medi	0.36	veterinarias	0.33
13	model	0.35	farmazia	0.31
14	reducziuns	0.35	survegli	0.31
Topic 19				
0	nasa	0.84	salt	0.62
1	nobel	0.8	clientas	0.58
2	zurbuchen	0.7	electricitad	0.57
3	premi	0.68	repower	0.56
4	musk	0.56	clients	0.52
5	one	0.52	cumbel	0.49
6	space	0.51	swisscom	0.49
7	scientific	0.48	rait	0.48
8	thomas	0.47	lincap	0.46
9	alfred	0.46	panna	0.46
10	academia	0.46	surcasti	0.44
11	scienzas	0.46	sutstaziun	0.43
12	univers	0.44	lelectricitad	0.43
13	elon	0.44	mobila	0.41
14	umanitad	0.43	clientella	0.4

Table 18: This compares all topic modeling results from the RTR corpus between both encoder models.