alokaili_2020_automatic_generation_of_topic_labels

Year

2020

Author(s)

Alokaili, Areej and Aletras, Nikolaos and Stevenson, Mark

Title

Automatic Generation of Topic Labels

Venue

SIGIR '20

Topic labeling

Fully automated

Focus

Primary

Type of contribution

Novel approach

Underlying technique

Sequence-to-sequence model (Sutskever et al., 2014, Cho et al., 2014)

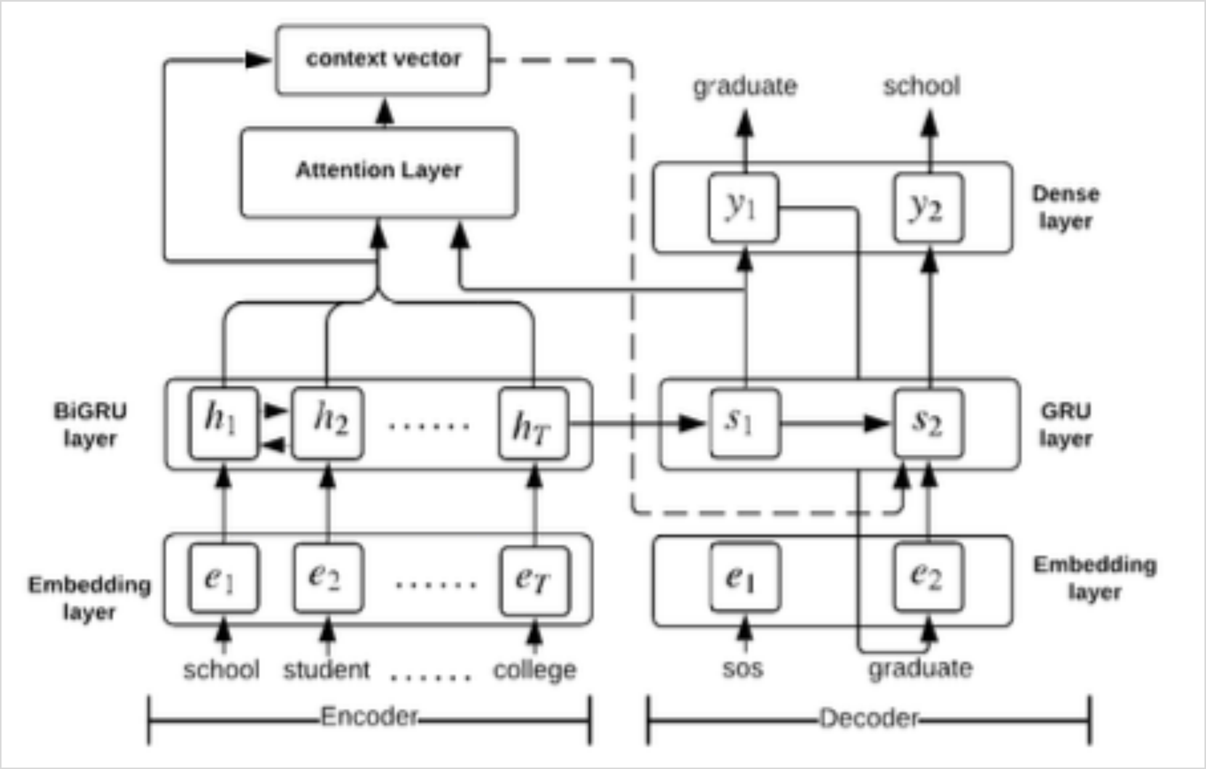
Topic labeling parameters

- **ENCODER**
  - Embedding layer:
    - INPUT: The topic terms.
    - OUTPUT: Term embeddings.

        - PARAMETER(S):
            - Dimension of the embedding: 300
    - Bidirectional GRU layer:
        - INPUT: Term embeddings
        - OUTPUT: Hidden state obtained by concatenating the encoded term information from the forward and backward GRU.
        - PARAMETER(S):
            - Nr. of units: 200
            - Dropout: 0.1
- **DECODER**
    - Embedding layer:
        - INPUT: SOS token at first time step, then output token generated at previous time step.
        - OUTPUT: Token embedding.
        - PARAMETER(S):
            - Dimension of the embedding: 300
    - GRU layer:
        - INPUT:
            - Embedding of the SOS token at first time step, then embedding of the token generated at previous time step.
            - Encoder hidden state at first time step, then decoder hidden state generated at previous time step.
            - Context vector computed for each target word (see attention layer).
        - OUTPUT: Hidden state at current time step.
        - PARAMETER(S):
            - Nr. of units: 200
            - Dropout: 0.1
    - Dense layer:
        - INPUT: Hidden state at current time step.
        - OUTPUT: Probability distribution over all vocabulary items (with highest prob. word chosen as label).
        - PARAMETER(S):
            - Activation function: softmax
- **ATTENTION LAYER (AND FEEDFORWARD NN)**
    - INPUT:
        - All encoder hidden states.
        - Decoder hidden state at previous time step.
    - OUTPUT: The context vector computed as a weighted sum over all encoder hidden states. Weights computed using an attention mechanism (giving an

higher weight to a specific state based on the value passed by the decoder).

- OTHER PARAMETER(S)
  - Optimiser: Adam with learning rate 0.001 and sparse categorical cross entropy loss.



Label generation

Label generation performed using a S2S model trained on a large synthetic dataset created using distant supervision
Akin to a traditional implementation of a Sequence-to-Sequence model with the addition of the context vector generated from the attention layer.

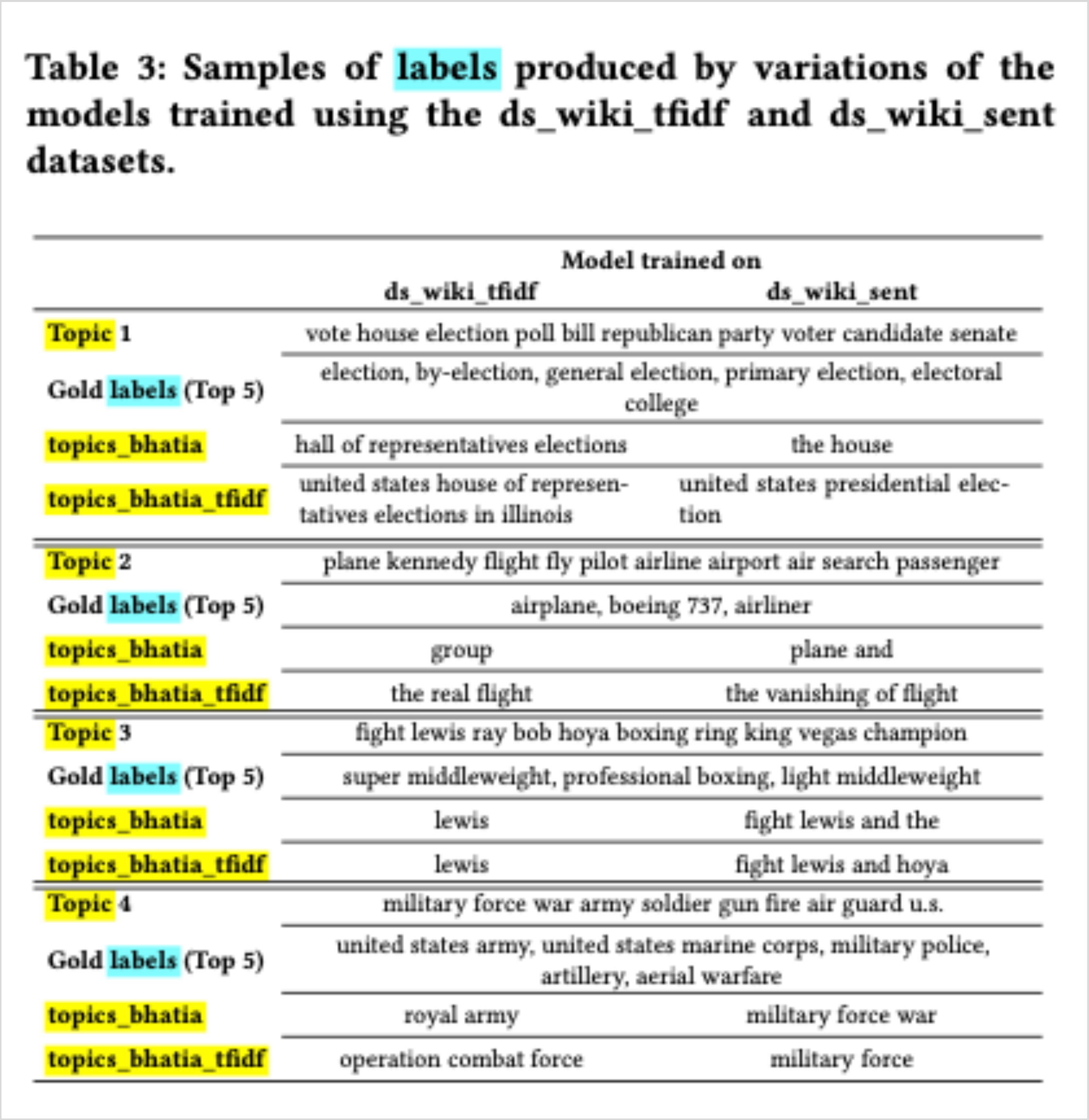


Table 3: Samples of labels produced by variations of the models trained using the ds_wiki_tfidf and ds_wiki_sent datasets.

	Model trained on	
	ds_wiki_tfidf	ds_wiki_sent
Topic 1	vote house election poll bill republican party voter candidate senate	
Gold labels (Top 5)	election, by-election, general election, primary election, electoral college	
topics_bhatia	hall of representatives elections	the house
topics_bhatia_tfidf	united states house of representatives elections in illinois	united states presidential election
Topic 2	plane kennedy flight fly pilot airline airport air search passenger	
Gold labels (Top 5)	airplane, boeing 737, airliner	
topics_bhatia	group	plane and
topics_bhatia_tfidf	the real flight	the vanishing of flight
Topic 3	fight lewis ray bob hoya boxing ring king vegas champion	
Gold labels (Top 5)	super middleweight, professional boxing, light middleweight	
topics_bhatia	lewis	fight lewis and the
topics_bhatia_tfidf	lewis	fight lewis and hoya
Topic 4	military force war army soldier gun fire air guard u.s.	
Gold labels (Top 5)	united states army, united states marine corps, military police, artillery, aerial warfare	
topics_bhatia	royal army	military force war
topics_bhatia_tfidf	operation combat force	military force

Motivation

"A limitation of these extractive approaches [labels from Wikipedia articles, from knowledge bases, from images] to label generation is that they are restricted to assigning labels that are found within the set of candidates.
This paper presents an alternative approach that does not suffer from this limitation.
It describes a neural-based model that automatically generates labels for topics in a single step, instead of retrieving and ranking candidates."

Topic modeling

- **Train data 1 and 2**
  - Distant supervision approach on Wikipedia articles.

Topic modeling parameters

- **Train data 1** (ds_wiki_tfidf)
    - Nr. of top TFIDF terms selected for article topic: 30
- **Train data 2** (ds_wiki_sent)
    - Nr. of initial article words selected for article topic: 30

Nr. of topics

- **Train data 1 and 2**
    - 300.000 topics (Train: 226,282, Test: 12,424, Validate:11,800)
- **Test data**
    - 219 topics

Label

- **Train data 1 and 2**
    - Single or multi-word label obtained from the Wikipedia article title used to generate the topic terms.
- **Test data**
    - Single or multi-word label obtained from Wikipedia titles.
- **Final results**
    - Single or multi-word label learned form train data.

Label selection

- **Test data**
    - Label (candidates) selected from a pool of 19 candidates using human assessors.

Label quality evaluation

Mean pairwise BERTScores (Zhang et al., 2019) between the topic's generated label and the (human generated) gold labels averaged across all topics.

BERTScore is a measure that computes the similarity be- tween predictions and references using contextual embeddings that has shown to have high correlation with human judgments.
Since BERTScore does not rely on exact matches between predicted and gold-standard labels, it is able to identify appropriate label words that do not appear in the gold labels.

Pairwise BERTScores between the topic's generated label *l* and the gold labels (*gold_l*1, …,*gold_ln*) is computed as follows:

$$score\_topic_t = \max_{i=[1,\dots,n]} BERTScore(l_t, gold\_l_{ti})$$

The model's overall score is the mean score over all topics:

$$score\_model = \frac{1}{T}\sum_{t=1}^{T} score\_topic_t$$

Comparison baselines

The labels generated by the proposed models were compared with two baselines:

- the top two terms, in terms of highest marginal probabilities, for a topic (Top-2 label)
- the top three terms (Top-3 label).

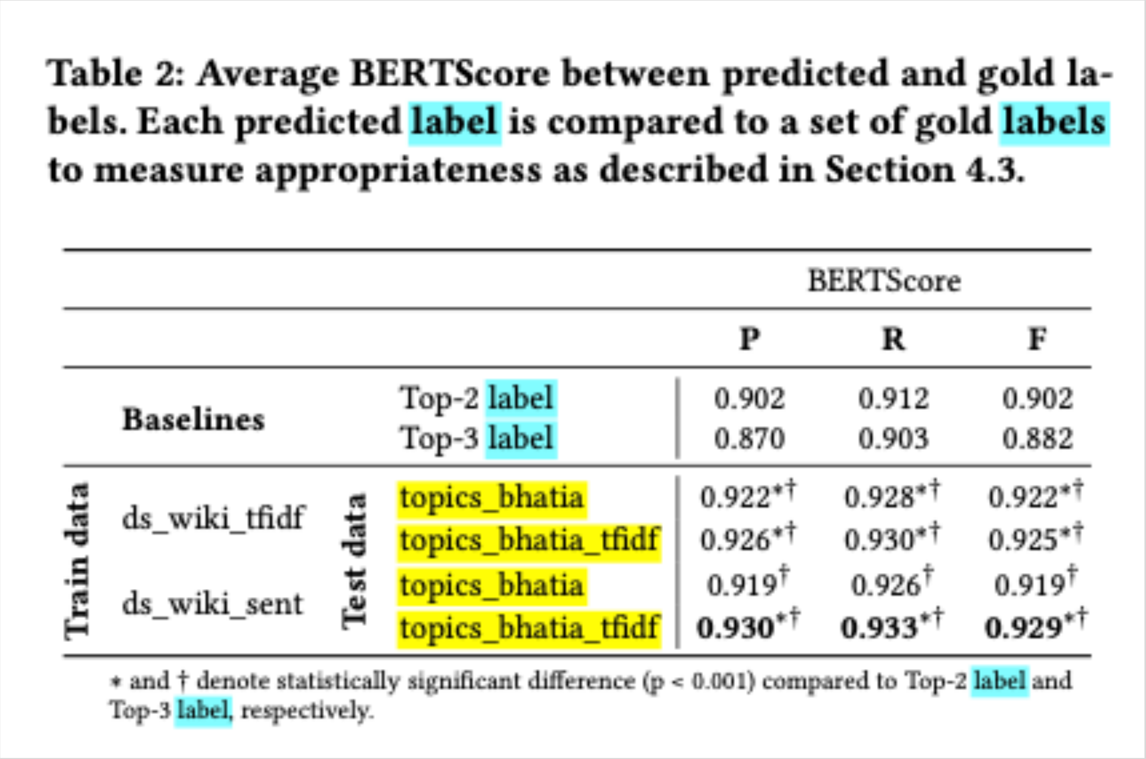


Table 2: Average BERTScore between predicted and gold labels. Each predicted label is compared to a set of gold labels to measure appropriateness as described in Section 4.3.

				BERTScore		
				P	**R**	**F**
Baselines			Top-2 label	0.902	0.912	0.902
			Top-3 label	0.870	0.903	0.882
Train data	ds_wiki_tfidf	**Test data**	topics_bhatia	0.922*†	0.928*†	0.922*†
			topics_bhatia_tfidf	0.926*†	0.930*†	0.925*†
	ds_wiki_sent		topics_bhatia	0.919†	0.926†	0.919†
			topics_bhatia_tfidf	**0.930*†**	**0.933*†**	**0.929*†**

* and † denote statistically significant difference ($p < 0.001$) compared to Top-2 label and Top-3 label, respectively.

Assessors

- **Test data**
  - CrowdFlower to collect human judgements.
  - Each candidate label rated by 10 annotators.

Domain

Domain (paper): Topic labeling

Domain (dataset):

- **Train data 1 and 2**
    - Miscellaneous (Wikipedia articles).
- **Test data**
    - blogs, books, news (wide-ranging topics from product reviews to religion to finance and entertainment)
    - PubMed (medical-domain specific)

Problem statement

Proposing using a sequence- to-sequence neural-based approach to generate topic labels.
Training the model over a new large synthetic dataset created using distant supervision.
Comparing the labels it generates to ones rated by humans.

Corpus

- **Train data 1 and 2** (ds_wiki_tfidf, ds_wiki_sent)
    - Origin: Wikipedia
    - Nr. of documents: 300.000

- **Test data** (topics_bhatia)
    - Origin: Bhatia et al., 2016
        - Blog articles from the Spinn3r blog dataset.
        - Books from the Internet Archive American Libraries collection.
        - New York Times articles from English Gigaword.
        - PubMed biomedical abstracts.
    - Nr of documents: ~207.000
        - Blogs: 120.000
        - Books: 1000
        - News: 29.000
        - PubMed: 77.000
- **(Test data 2)** (topics_bhatia_tfidf)
    - Extended version of topics_bhatia that includes 20 additional terms for each topic. These additional terms were added to the 10 from topics_bhatia so that each topic consists of 30 terms
    - Additional terms were identified by finding documents associated with each topic and choosing the 20 with the highest TFIDF scores.
    - Consequently suitable documents were identified by computing cosine similarity

between the topic terms and documents using word embeddings.

Document

/

Pre-processing

Removal of numbers, special characters, rare terms and stop words (Stop words not removed from headlines)

```
@inproceedings{alokaili_2020_automatic_generation_of_topic_labels,
author = {Alokaili, Areej and Aletras, Nikolaos and Stevenson, Mark},
title = {Automatic Generation of Topic Labels},
year = {2020},
isbn = {9781450380164},
publisher = {Association for Computing Machinery},
address = {New York, NY, USA},
url = {https://doi.org/10.1145/3397271.3401185},
doi = {10.1145/3397271.3401185},
abstract = {Topic modelling is a popular unsupervised method for identifying
the underlying themes in document collections that has many applications in
information retrieval. A topic is usually represented by a list of terms ranked
by their probability but, since these can be difficult to interpret, various
approaches have been developed to assign descriptive labels to topics. Previous
work on the automatic assignment of labels to topics has relied on a two-stage
approach: (1) candidate labels are retrieved from a large pool (e.g. Wikipedia
article titles); and then (2) re-ranked based on their semantic similarity to
the topic terms. However, these extractive approaches can only assign candidate
labels from a restricted set that may not include any suitable ones. This paper
proposes using a sequence-to-sequence neural-based approach to generate labels
that does not suffer from this limitation. The model is trained over a new
large synthetic dataset created using distant supervision. The method is
evaluated by comparing the labels it generates to ones rated by humans.},
booktitle = {Proceedings of the 43rd International ACM SIGIR Conference on
Research and Development in Information Retrieval},
pages = {1965–1968},
numpages = {4},
keywords = {topic modeling, neural network, topic representation},
location = {Virtual Event, China},
series = {SIGIR '20}
}

@article{sutskever_2014_sequence_to_sequence_learning_with_neural_networks,
  abstract = {Deep Neural Networks (DNNs) are powerful models that have
achieved excellent performance on difficult learning tasks. Although DNNs work
well whenever large labeled training sets are available, they cannot be used to
map sequences to sequences. In this paper, we present a general end-to-end
approach to sequence learning that makes minimal assumptions on the sequence
structure. Our method uses a multilayered Long Short-Term Memory (LSTM) to map
the input sequence to a vector of a fixed dimensionality, and then another deep
LSTM to decode the target sequence from the vector. Our main result is that on
an English to French translation task from the WMT'14 dataset, the translations
produced by the LSTM achieve a BLEU score of 34.8 on the entire test set, where
the LSTM's BLEU score was penalized on out-of-vocabulary words. Additionally,
the LSTM did not have difficulty on long sentences. For comparison, a phrase-
based SMT system achieves a BLEU score of 33.3 on the same dataset. When we
used the LSTM to rerank the 1000 hypotheses produced by the aforementioned SMT
system, its BLEU score increases to 36.5, which is close to the previous best
result on this task. The LSTM also learned sensible phrase and sentence
representations that are sensitive to word order and are relatively invariant
to the active and the passive voice. Finally, we found that reversing the order
of the words in all source sentences (but not target sentences) improved the
LSTM's performance markedly, because doing so introduced many short term
dependencies between the source and the target sentence which made the
optimization problem easier.},
  author = {Ilya Sutskever and Oriol Vinyals and Quoc V. Le},
  date-added = {2023-02-27 10:17:56 +0100},
  date-modified = {2023-02-27 10:17:56 +0100},
  eprint = {1409.3215},
  month = {09},
  title = {Sequence to Sequence Learning with Neural Networks},
  url = {https://arxiv.org/pdf/1409.3215.pdf},
  year = {2014},
  bdsk-url-1 = {https://arxiv.org/pdf/1409.3215.pdf},
  bdsk-url-2 = {https://arxiv.org/abs/1409.3215}}

@inproceedings{cho_2014_learning_phrase_representations_using_rnn_encoder_decoder_for_statistical_machine_translation,
    title = "Learning Phrase Representations using {RNN} Encoder{--}Decoder for Statistical Machine Translation",
    author = {Cho, Kyunghyun  and
      van Merri{\"e}nboer, Bart  and
      Gulcehre, Caglar  and
      Bahdanau, Dzmitry  and
      Bougares, Fethi  and
      Schwenk, Holger  and
      Bengio, Yoshua},
    booktitle = "Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing ({EMNLP})",
    month = oct,
    year = "2014",
    address = "Doha, Qatar",
    publisher = "Association for Computational Linguistics",
    url = "https://aclanthology.org/D14-1179",
    doi = "10.3115/v1/D14-1179",
    pages = "1724--1734",
}

@inproceedings{bhatia_2016_automatic_labelling_of_topics_with_neural_embeddings,
    title = "Automatic Labelling of Topics with Neural Embeddings",
    author = "Bhatia, Shraey  and
      Lau, Jey Han  and
      Baldwin, Timothy",
    booktitle = "Proceedings of {COLING} 2016, the 26th International Conference on Computational Linguistics: Technical Papers",
    month = dec,
    year = "2016",
    address = "Osaka, Japan",
    publisher = "The COLING 2016 Organizing Committee",
    url = "https://aclanthology.org/C16-1091",
    pages = "953--963",
    abstract = "Topics generated by topic models are typically represented as
```

```
list of terms. To reduce the cognitive overhead of interpreting these topics
for end-users, we propose labelling a topic with a succinct phrase that
summarises its theme or idea. Using Wikipedia document titles as label
candidates, we compute neural embeddings for documents and words to select the
most relevant labels for topics. Comparing to a state-of-the-art topic
labelling system, our methodology is simpler, more efficient and finds better
topic labels.",
}


@article{zhang_2019_bertscore_evaluating_text_generation_with_bert,
  abstract = {We propose BERTScore, an automatic evaluation metric for text
generation. Analogously to common metrics, BERTScore computes a similarity
score for each token in the candidate sentence with each token in the reference
sentence. However, instead of exact matches, we compute token similarity using
contextual embeddings. We evaluate using the outputs of 363 machine translation
and image captioning systems. BERTScore correlates better with human judgments
and provides stronger model selection performance than existing metrics.
Finally, we use an adversarial paraphrase detection task to show that BERTScore
is more robust to challenging examples when compared to existing metrics.},
  author = {Tianyi Zhang and Varsha Kishore and Felix Wu and Kilian Q.
Weinberger and Yoav Artzi},
  date-added = {2023-02-27 12:30:36 +0100},
  date-modified = {2023-02-27 12:30:36 +0100},
  eprint = {1904.09675},
  month = {04},
  title = {BERTScore: Evaluating Text Generation with BERT},
  url = {https://arxiv.org/pdf/1904.09675.pdf},
  year = {2019},
  bdsk-url-1 = {https://arxiv.org/pdf/1904.09675.pdf},
  bdsk-url-2 = {https://arxiv.org/abs/1904.09675}}
```

#Thesis/Papers/Initial