RePEAtO: Relational Paragraph-level Embeddings for Article Outlining

PRINCETON UNIVERSITY

Michael Tang, Evan Dogariu, Jiatong Yu

Introduction

- Past work in NLP representation learning has focused heavily on word-level, and more recently, sentence-level embeddings. **Paragraphs** encode the flow of information across individual **sentences** when illustrating more complex ideas – thus, we are interested in exploring ways of encoding at the paragraph level.
- We propose the task of *article outlining*, predicting the nested outline of an article given only the raw paragraph sequence. We will focus on learning the structure rather than the names of the headings.
- Possible applications include:
  - generating outlines of freeform notes
  - reconstructing structure of audio transcriptions
  - help with retriever systems via subheadings

In general, we hope paragraph-level understanding will help models interact with text in more human-like ways.

Related Work

- Any approach to this problem has to involve embedding of the paragraphs into a useful latent space: this can be done with well known methods like a simple recurrent or transformer architecture with various training objectives (e.g. Doc2Vec [2]).
- The task on determining the relationship between paragraphs lends itself to contrastive learning of sequential data, which is explored at the sentence level in SimCSE [3], among others.
- We require the ability to use learned embeddings to construct a hierarchical structure; a possible dynamic programming approach to segmentation using embedding similarities is explored in [1].

Methods

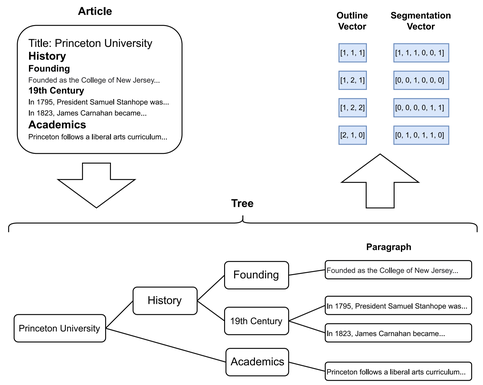

Figure 3: Article tree and vector forms

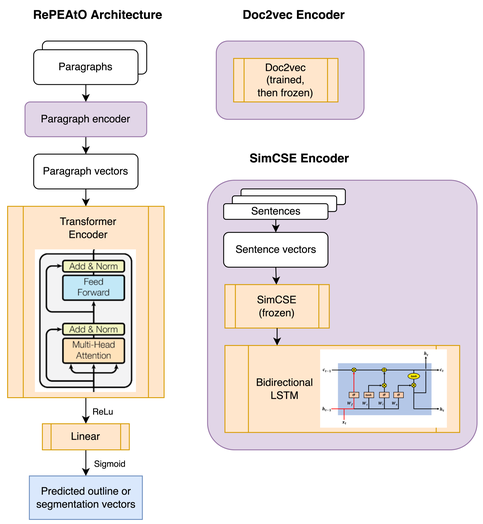

Figure 4: End-to-end model architecture

We formulate the problem as the prediction of the *heading tree* — the nested n-ary tree of headings — from a list of raw paragraphs in an article. For training, we formulate the output as a segmentation vector, which encodes whether a paragraph starts or ends a new section at each depth.

We propose 3 different architectures and techniques. All have an **encoder/decoder** structure, where we encode each article's paragraphs into some embedding space and then predict the heading tree from the sequence of embedded paragraphs. We attempt each technique with a variety of embeddings (doc2vec, averaged/projected SimCSE, etc.)

1. **Greedy Decoder -** Given paragraph embeddings, we compute pairwise similarities and recursively group together pairs above a tuned threshold. We build the tree from the bottom up based on similarity groupings.
2. **Recursive MLP -** Given paragraph embeddings, we use a context window to encode information about a target paragraph. We pass this into a MLP to predict whether the paragraph divides a subheading at a given depth. We use this recursively to build the tree from the top down via divisions.
3. **End to End -** Given paragraph embeddings, we apply a Transformer to create an output sequence of a certain format (outlines or segmentation vectors). We convert this representation to the tree directly.

We train these models using the Huggingface Wikitext dataset, containing 30,000 Wikipedia articles.

Results

```
[heading]
   Dan Dugan ( born March 20 , 1943 ) is a...
   In his youth , Dugan was fascinated by ...
   Dugan first recorded sounds in the late...
[heading]
   Daniel W. Dugan was born in Los Angeles...
[heading]
   Dugan changed from lighting design to s...
   Dugan designed sound for three regional...
   When Margrit Mondavi founded the Mondav...
   Dugan occasionally delivered speeches a...
[heading]
   While designing sound for the musical H...
   Though the algorithm was good , the ref...
   " I was messing around with logarithmic...
   Dugan licensed this more practical syst...
   In the late 1980s , Dugan developed a g...
   Dugan 's original 1974 patent expired i...
   In September 2006 , Dugan produced the ...
   In February 2011 , Dugan demonstrated a...
[heading]
   Dugan made his first sound effects reco...
   Dugan and his wife Sharon Perry , the N...
   " There are three potential values in s...
   In 2006 , Dugan assisted a group of res...
[heading]
   In 1998 an organization he co @-@ found...
   As co @-@ founder and Secretary of PLAN...
```

Figure 1: Ground truth tree for example article "Dan Dugan (audio engineer)"

```
[heading]
  [heading]
    [heading]
      [heading]
         Dan Dugan ( born March 20 , 1943 ) is a...
         In his youth , Dugan was fascinated by ...
         Dugan first recorded sounds in the late...
       Daniel W. Dugan was born in Los Angeles...
      [heading]
         Dugan changed from lighting design to s...
         Dugan designed sound for three regional...
         When Margrit Mondavi founded the Mondav...
    [heading]
       Dugan occasionally delivered speeches a...
      [heading]
         While designing sound for the musical H...
         Though the algorithm was good , the ref...
         " I was messing around with logarithmic...
       Dugan licensed this more practical syst...
      [heading]
         In the late 1980s , Dugan developed a g...
         Dugan 's original 1974 patent expired i...
      [heading]
         In September 2006 , Dugan produced the ...
         In February 2011 , Dugan demonstrated a...
       Dugan made his first sound effects reco...
       Dugan and his wife Sharon Perry , the N...
       " There are three potential values in s...
       In 2006 , Dugan assisted a group of res...
       In 1998 an organization he co @-@ found...
       As co @-@ founder and Secretary of PLAN...
```

Figure 2: Predicted tree for example article "Dan Dugan (audio engineer)" from Greedy-Doc2Vec

	Greedy-Doc2Vec	Greedy-SimCSE	Recursive	End-to-End
LCA distance	30.04	36.77	32.43	84.34

- We measure performance (evaluate produced trees against ground truths) via **LCA distance**. We compute the mean squared difference between all-pairs LCA distances averaged over predicted and ground truth trees: $\sum_{i \neq j} (\text{LCA}_{ij}^{\hat{y}} - \text{LCA}_{ij}^{y})^2$.
- The **End to End** models perform the worst. They predict the same segmentation sequence for any input. After investigating, we found that this correlates strongly with the mean of the segmentation vectors over the training set. This behavior persists throughout various changes in architecture, hyperparameters, overfitting conditions, etc. We attempting to correct for this via normalization by mean and standard deviation but it continues to predict a constant output.

Conclusion

We approach a novel problem (article outlining) with a simple framework (encoder/decoder) made of composable parts. We experiment and ablate different techniques for embedding and decoding, along with different sequential tree representations. We also construct an evaluation metric (LCA distance) that allows for unified analysis of different methods. Overall, we lay the groundwork for more research on this topic, including *custom supervised contrastive learning paragraph embeddings*.

References

[1] Chong Wang, Yining Wang, Po-Sen Huang, Abdelrahman Mohamed, Dengyong Zhou, and Li Deng. Sequence modeling via segmentations. In *International Conference on Machine Learning*, pages 3674–3683. PMLR, 2017.

[2] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196. PMLR, 2014.

[3] Tianyu Gao, Xingcheng Yao, and Danqi Chen. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*, 2021.