TITLE

Pre-Defense MPhil assessed by dr. Meike Morren & dr. Jonne Guyt

Finn-Ole Höner; Supervisors: prof. Dennis Fok & prof. Bas Donkers

June 28, 2024

One-Slide Summary

- New method to calculate document summaries

Table 1: Overview of existing methods to summarize documents.

Method	Origin	Reference	Deterministic?	Generation?	Type?
BERT `[CLS]` token	Next-sentence prediction task	Devlin et al. (2018)	✓	×	Numeric
Pooled Word Embeddings	Aggregation of token information	Tomáš Mikolov et al. (2013) Shen et al. (2018)	✓	×	Numeric
(Optimized) Prompt engineering	Emergent capability of LLM	Khattab et al. (2023); Huang and Chen (n.d.)	×	✓	Textual
Reverse engineered document summaries	Maximize the likelihood to re-generate the focal document	*Proposed method*	✓	✓	Numeric

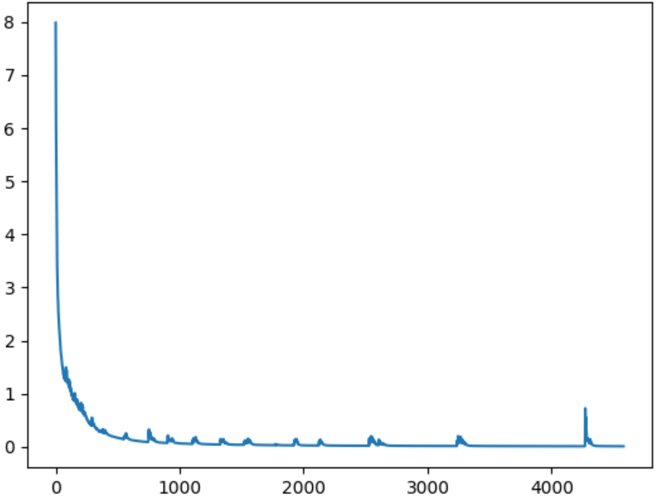
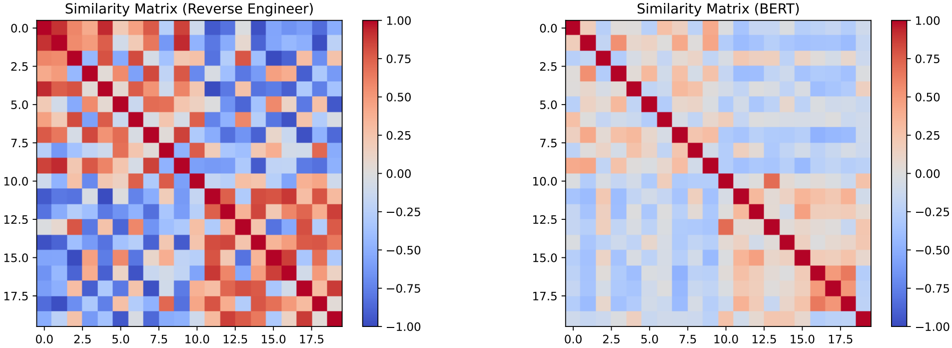


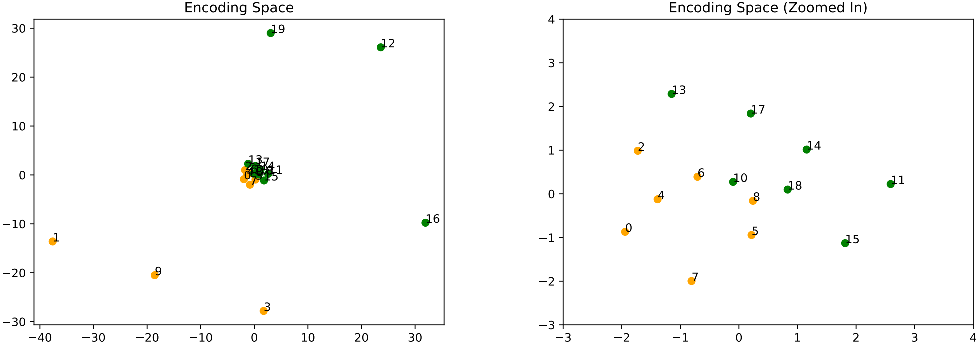
Figure 1: Training of embeddings



(a) SE: 2 Factors with LayerNorm.

(b) BERT

Figure 2: Correlation matrices along the claims.



(a) All points

(b) Zoom-in

Figure 3: Encoding space of the advertising claims. Numbered points are locations of claims that are part of the training data, colored dots are points where we generate one of the claims that are part of the training data, and red crosses are locations where we generate a claim that ends with the eos-token, but is not part of the training data. Whitespace marks coordinates where we generate a string that does not contain an eos

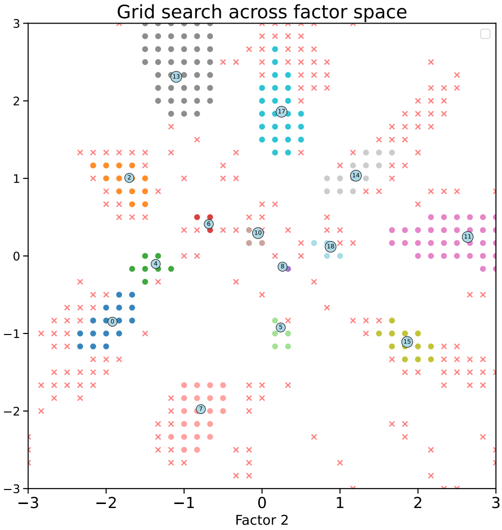


Figure 4: Exploration of Encoding Space

Table 2: Example for similar tokens

	3rd	2nd	Choice	Prob. 3rd	Prob. 2nd	Prob. Choice
0	See	Form	Experience	0.009	0.014	0.944
1	20	25	50	0.002	0.004	0.976
2	%.	.	%	0.001	0.002	0.994
3	better	More	more	0.001	0.002	0.996
4	shine	protective	visible	0.002	0.002	0.979
5	glow	light	shine	0.001	0.002	0.983
6	when	by	after	0.002	0.004	0.987
7	a	well	just	0.001	0.002	0.990
8	two	a	one	0.002	0.002	0.990
9	shine	usage	use	0.000	0.003	0.991
10	,	!	.	0.000	0.001	0.998
11	.	*space*	*eos-token*	0.000	0.002	0.994

Table 3: Example for similar tokens

	Weight	Generated String
0	0/20	Unlock the secret to hair that shines from within, reflecting your inner glow. *eos-token*
1	1/20	Unlock the secret to hair that shining from within, your bedroom. *eos-token*
2	2/20	Unlock the secret to black metal's glow. *multiple eos-token*
3	3/20	Unlock the secret to blackened nails that shine from the shine. *eos-token*
4	4/20	Un to of for the team members upon *linebreak* Moderation of the, that
5	...	...
17	17/20	Rediscover the joy of hair that's at the-corrects life with a healthy
18	18/20	Rediscover the joy of hair that beams with inner vibrancy. *eos-token*
19	19/20	Rediscover the joy of hair that beams with inner vibrancy. *eos-token*

Algorithm 1 Training of Single Summary Embeddings

```
i ← 0
ε ← 0.01
s ← Initialization(·)
while True do
    l^(i) ← L(s | t_1, ..., t_T)
    ∇_s^(i) l ← ComputeGradient(l^(i))
    s^(i+1) l ← Optimizer(s^(i), ∇_s^(i) l)
    i ← i + 1
    if l^(i) < ε then
        break
    end if
end while
```

Where:

$i \leftarrow 0$

$\epsilon \leftarrow 0.01$

$\mathbf{s} \leftarrow \text{Initialization}(\cdot)$

while *True* **do**

  $l^{(i)} \leftarrow \mathcal{L}(\mathbf{s} \mid t_1, \ldots, t_T)$

  $\nabla_{\mathbf{s}}^{(i)} l \leftarrow \text{ComputeGradient}\left(l^{(i)}\right)$

  $\mathbf{s}^{(i+1)} l \leftarrow \text{Optimizer}\left(\mathbf{s}^{(i)}, \nabla_{\mathbf{s}}^{(i)} l\right)$

  $i \leftarrow i + 1$

  if $l^{(i)} < \epsilon$ **then**

    break

  end if

end while

Algorithm 2 Training of Summary Embeddings based on factor model

$i \leftarrow 0$

$\epsilon \leftarrow 0.01$

$\mathsf{W}^{Encoder}_{(i)}, \mathsf{W}^{Decoder}_{(i)}, \mathsf{B}^{Encoder}_{(i)}, \mathsf{B}^{Decoder}_{(i)} \leftarrow \text{Initialization}(\cdot)$

while *True* **do**

  $\mathsf{S} \leftarrow \left(\mathsf{W}^{Encoder}_{(i)} + \mathsf{B}^{Encoder}_{(i)}\right)\mathsf{W}^{Decoder}_{(i)} + \mathsf{B}^{Decoder}_{(i)}$

  $l_{(i)} \leftarrow \mathcal{L}_D(\mathsf{S})$

  $\nabla_{(i)} \leftarrow \text{ComputeGradient}\left(l^{(i)}\right)$

  $\mathsf{W}^{Encoder}_{(i+1)}, \mathsf{W}^{Decoder}_{(i+1)}, \mathsf{B}^{Encoder}_{(i+1)}, \mathsf{B}^{Decoder}_{(i+1)} \leftarrow \text{Optimizer}\left(\mathsf{W}^{Enc.,Dec.}_{(i+1)}, \mathsf{B}^{Enc.,Dec.}_{(i+1)}, \nabla_{(i)}\right)$

  $i \leftarrow i + 1$

  if $l_{(i)} < \epsilon$ **then**

    break

  end if

end while

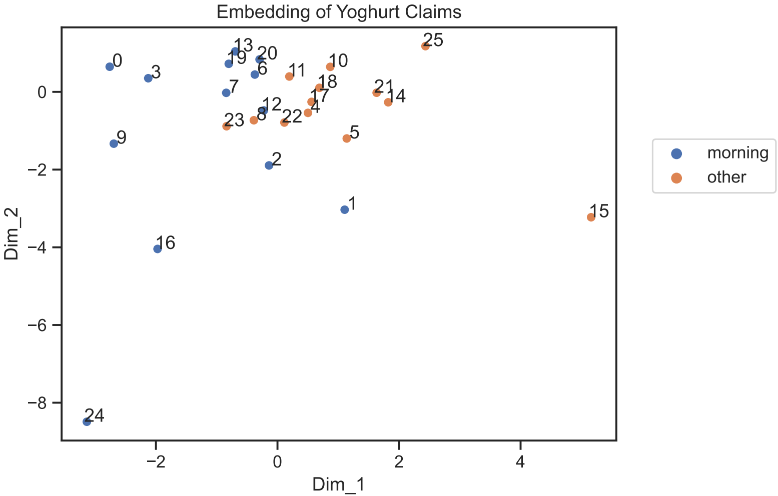


Figure 5: Document Summaries

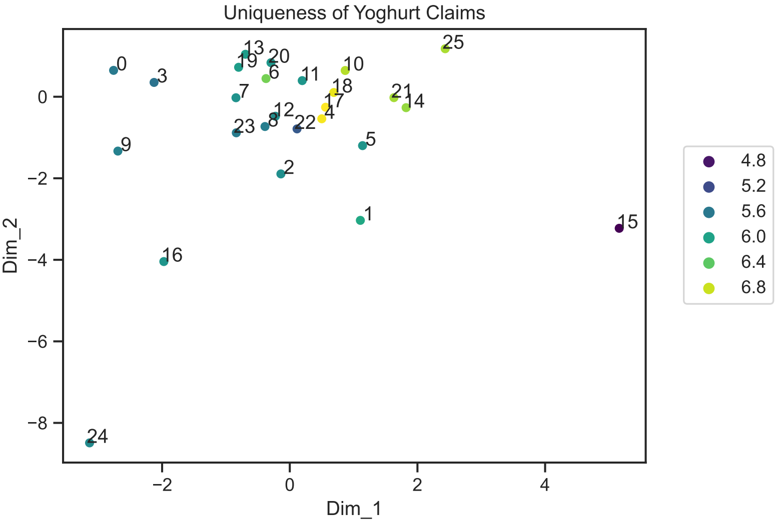


Figure 6: Document Summaries

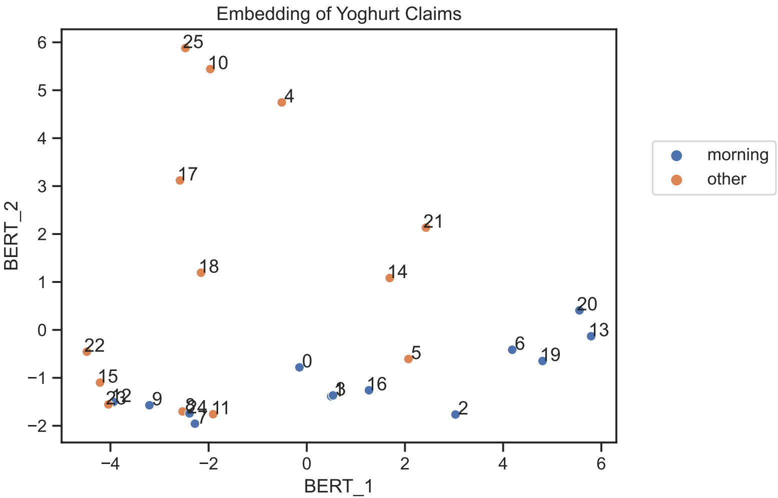


Figure 7: Document Summaries

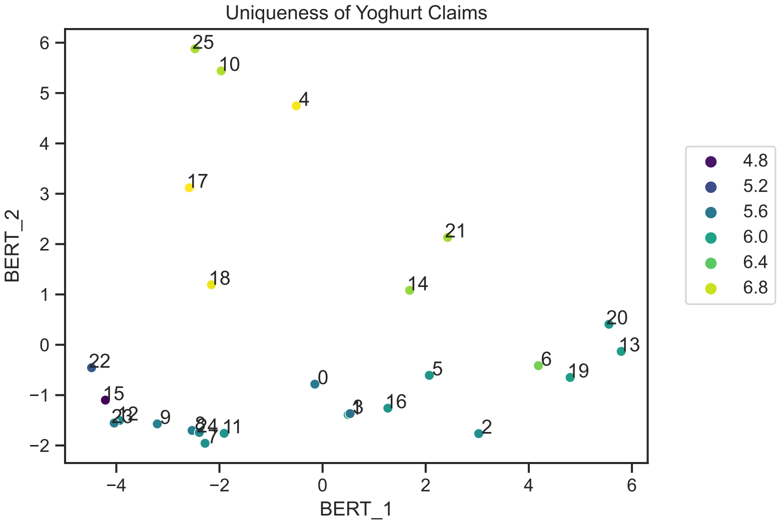


Figure 8: Document Summaries

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." https://doi.org/10.48550/ARXIV.1810.04805.

Huang, Zihao, and Tao Chen. n.d. "Enhanced News Summarization Using Large Language Models and Advanced Prompt Engineering."

Khattab, Omar, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, et al. 2023. "DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines." arXiv. https://arxiv.org/abs/2310.03714.

References ii

Shen, Dinghan, Guoyin Wang, Wenlin Wang, Martin Renqiang Min, Qinliang Su, Yizhe Zhang, Chunyuan Li, Ricardo Henao, and Lawrence Carin. 2018. "Baseline Needs More Love: On Simple Word-Embedding-Based Models and Associated Pooling Mechanisms." https://doi.org/10.48550/ARXIV.1805.09843.

Tomáš Mikolov, Tomas Mikolov, Ilya Sutskever, Ilya Sutskever, Kai Chen, Kai Chen, Kai Chen, et al. 2013. "Distributed Representations of Words and Phrases and Their Compositionality" 26 (December): 3111–19.