# Employing Bert Embeddings for Customer Segmentation and Translation Matching

## SemaNLP 2020

Tim vor der Brück
tim.vorderbrueck@hslu.ch
Lucerne University of Applied Sciences and Arts
Switzerland

October 2020

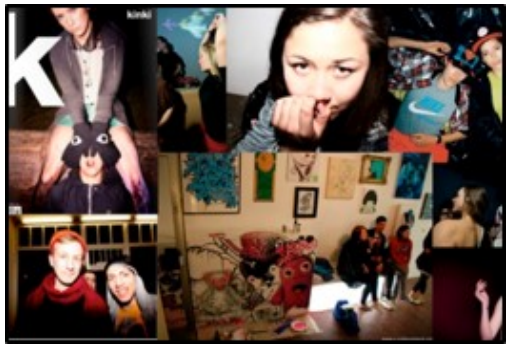
# CV

- 2001: Diploma Degree at Saarland University
- 2001-2003: Scientific Assistant at the German Research Center for Artificial Intelligence (DFKI)
- 2003-2006: Scientist at the German Meteorological Service
- 2006-2010: PhD Student at the Distance University of Hagen
- 2011-2015: Postdoc at Goethe University Frankfurt am Main
- 2015-now: Research Associate at Lucerne University of Applied Science and Arts
- 2019-now: Lecturer at FFHS (Fernfachhochschule Schweiz)

# Task: assign people to marketing target groups

- Our business partner operates a Website where he organizes several contests
- in these contests, people can win things like bicycles, MacBook, pair of sneakers
- For that, people had to come up with a short description what to do with their prize, or what they want to do in their dream holiday
- The participants of the contests were matched to one of 6 target groups

# Target Groups


NITRO CIRCUS


Keywords

Keywords


Keywords


Keywords


Keywords


Keywords

Keywords


Keywords


Keywords


Keywords


Keywords

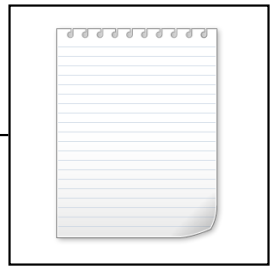
Keywords

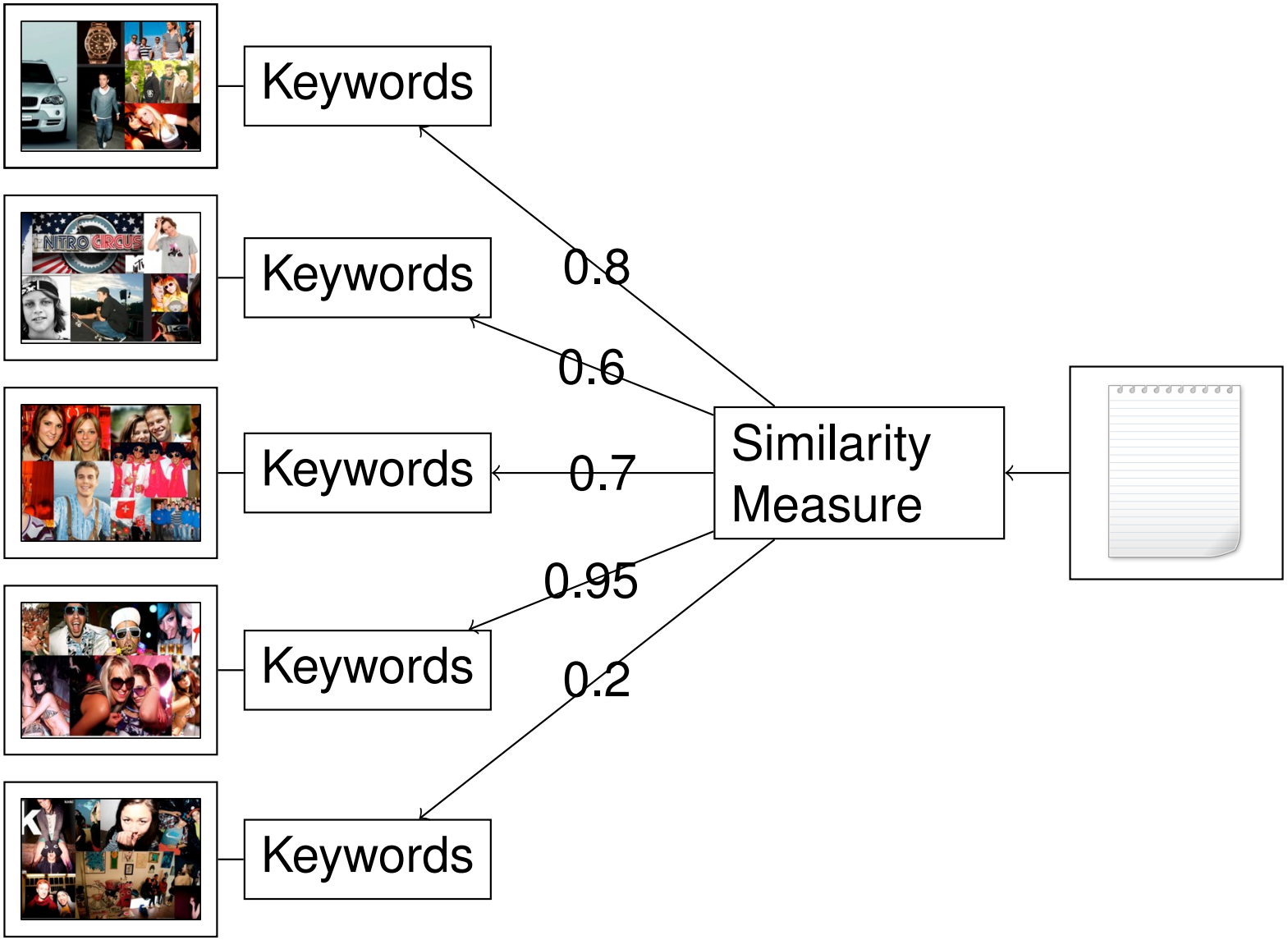

Keywords

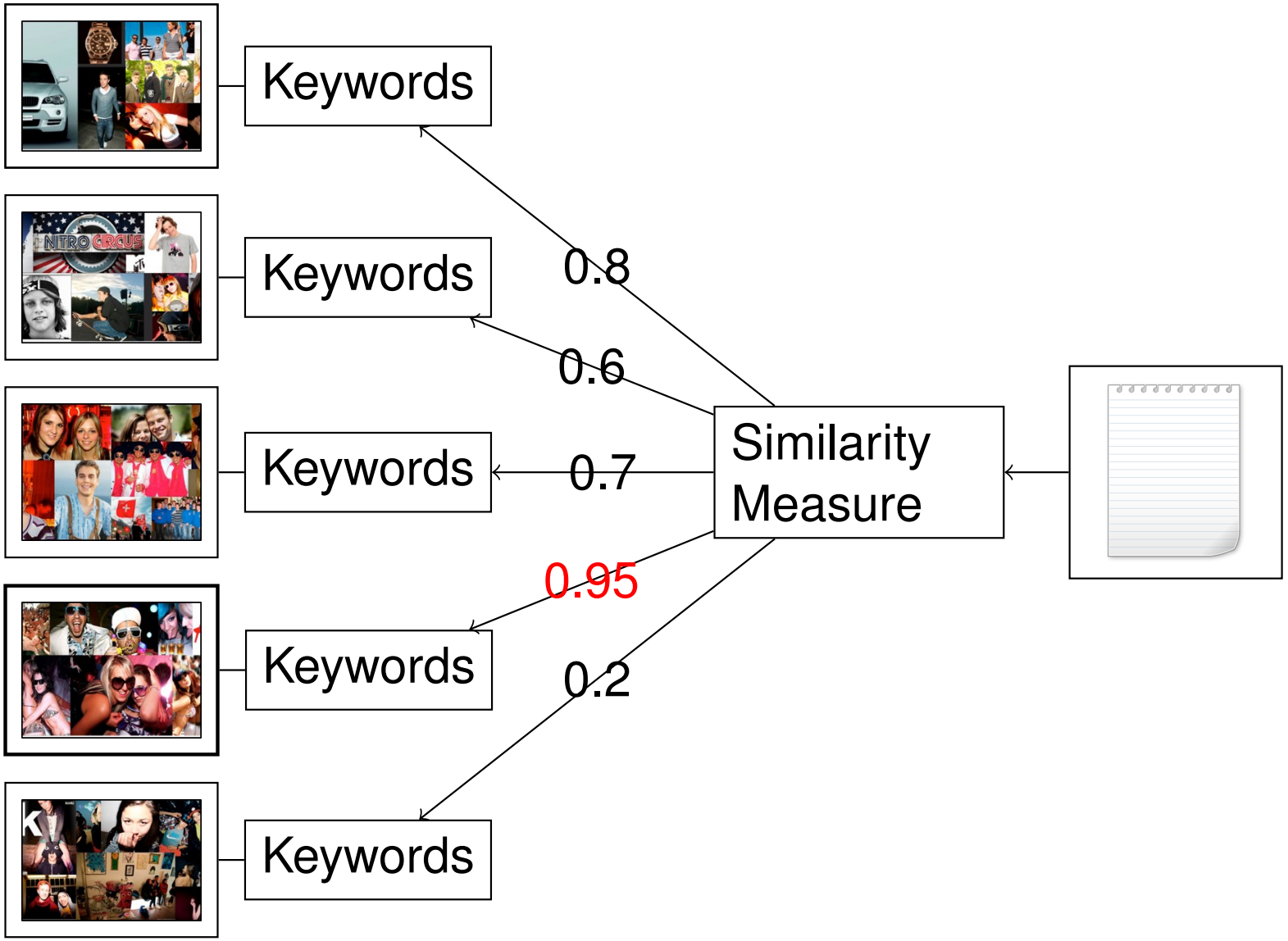

Keywords


Keywords

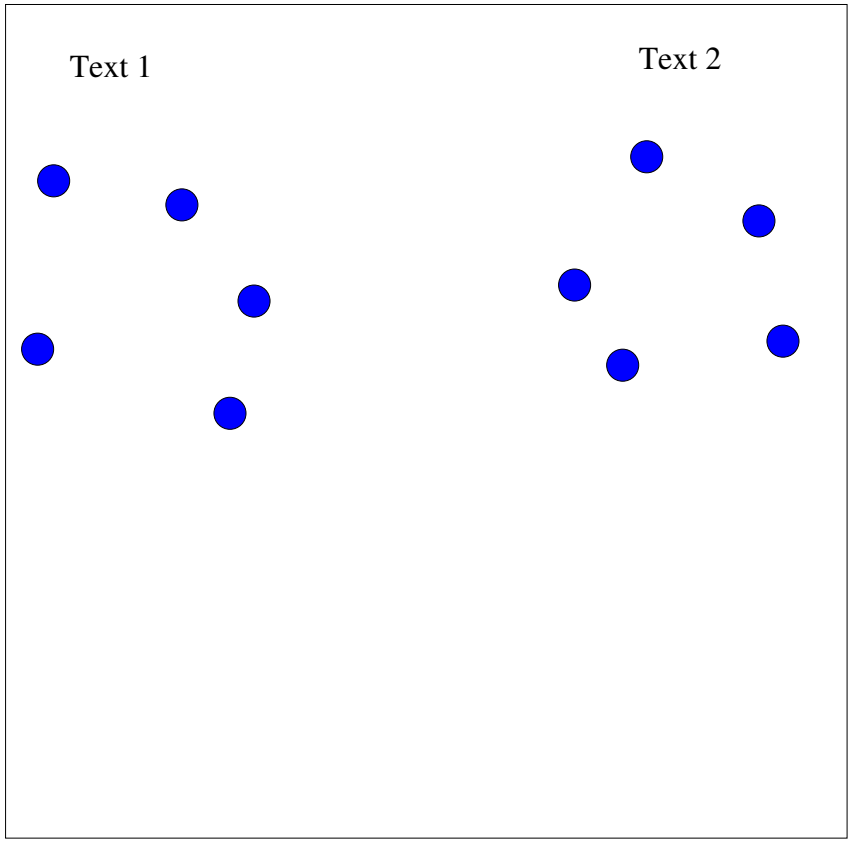
Similarity Measure



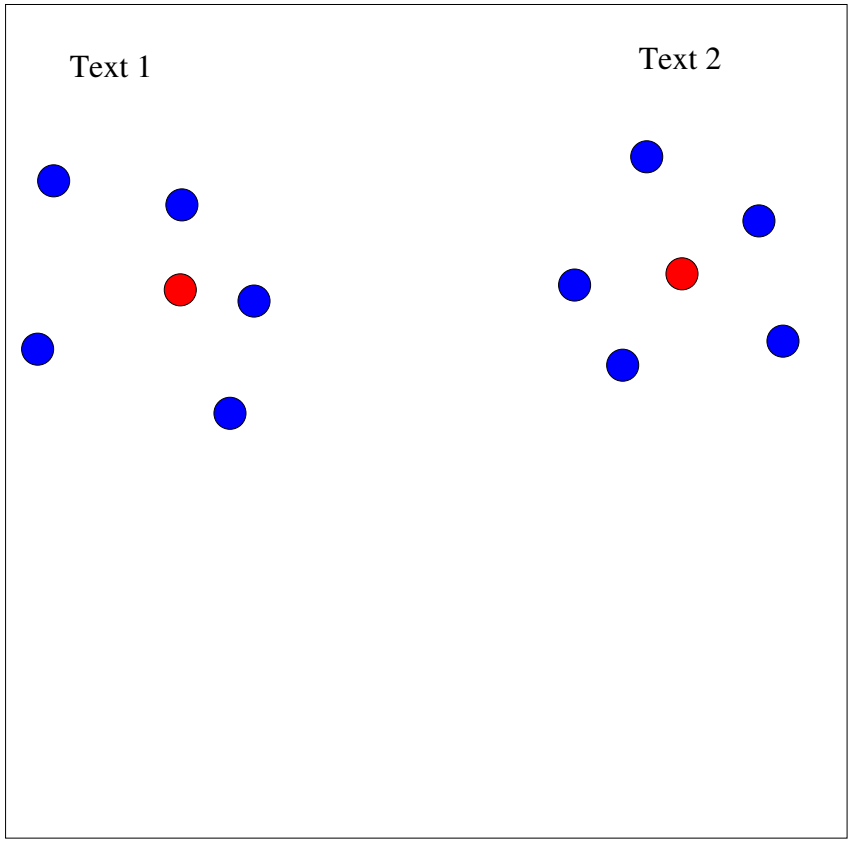

Keywords
Keywords
Keywords
Keywords
Keywords
0.8
0.6
0.7
0.95
0.2
Similarity Measure


Keywords
Keywords
Keywords
Keywords
Keywords
0.8
0.6
0.7
0.95
0.2
Similarity Measure

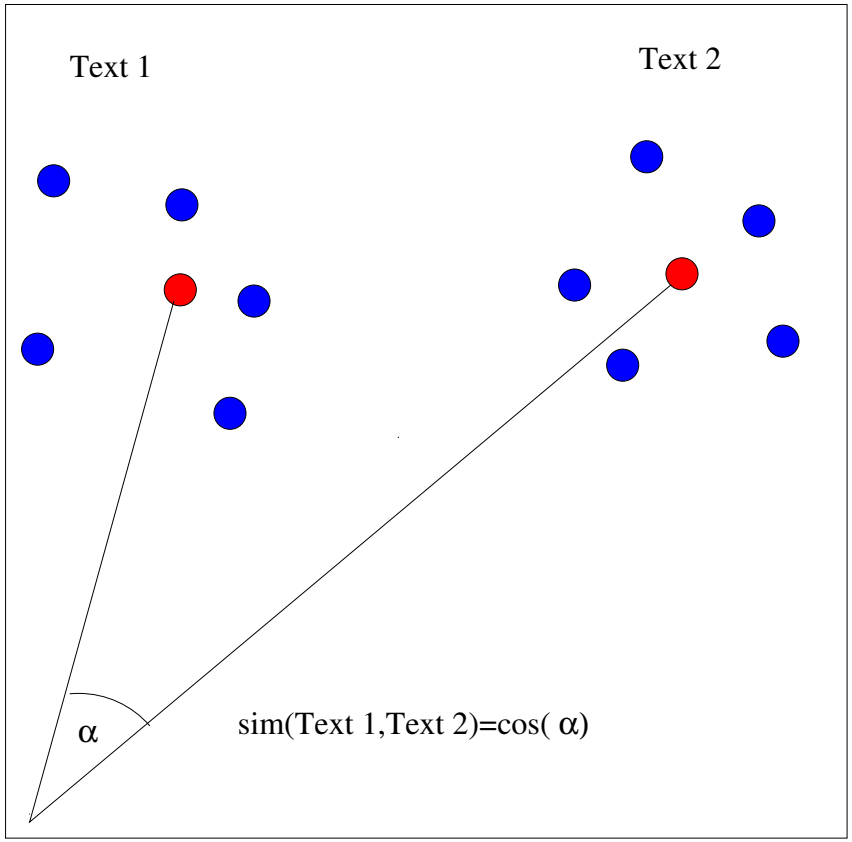
# Standard approach - Centroid of Embeddings (CE)



# Standard approach - Centroid of Embeddings (CE)



# Standard approach - Centroid of Embeddings (CE)



# Standard approach (CE)

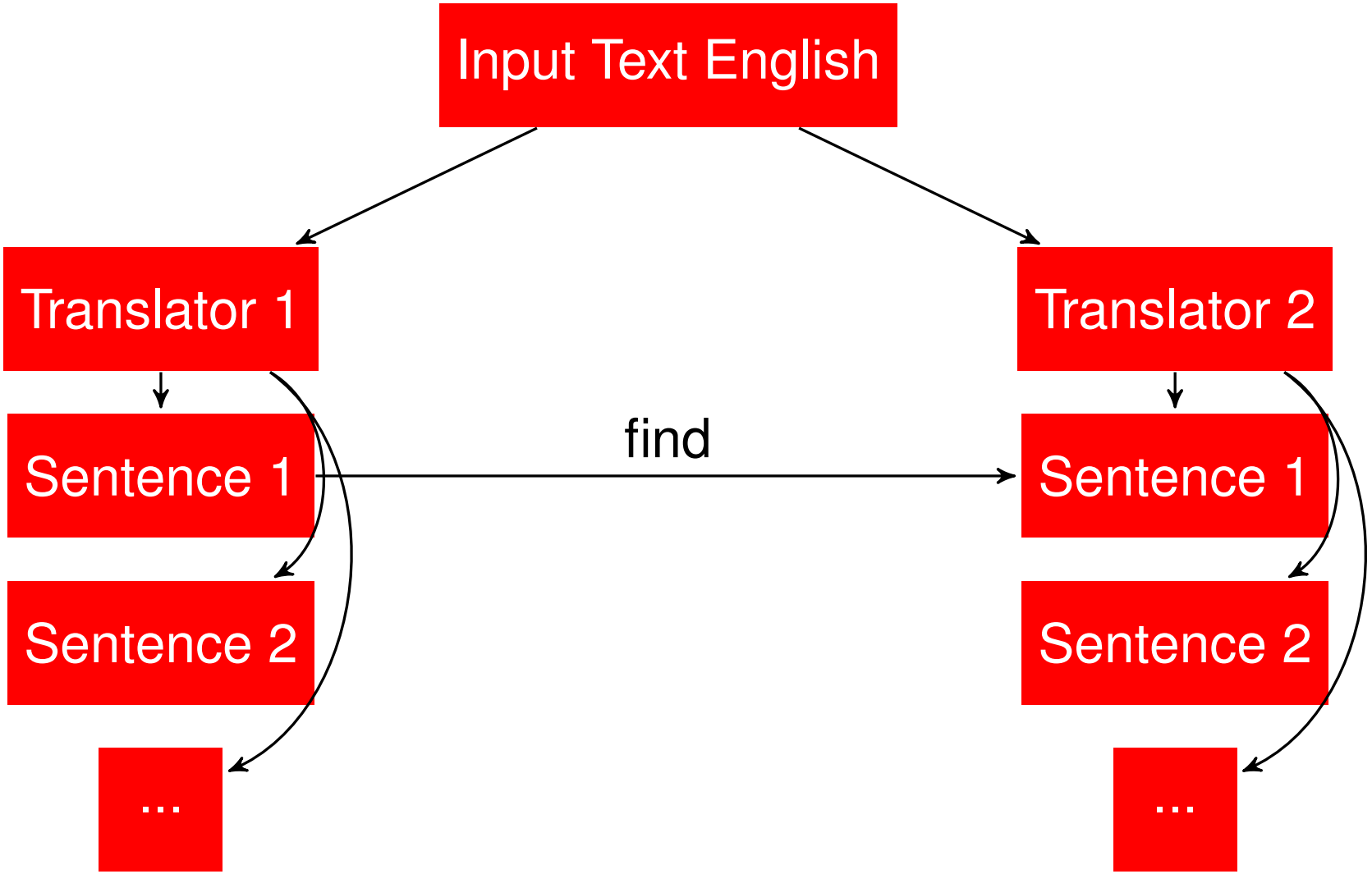
Task: Estimate semantic similarity of text *s* and text *t*

- Compute word embeddings for all words occurring in text *s* and *t*
- Compute the two centroids *cs* and *ct* of the word embeddings
- Similarity is given by: $cos(\angle(cs, ct))$

# Translation Matching

- One text (The Purloined Letter from Edgar Allen Poe) was translated by two independent translators into German
- Task: Matching one sentence of the first translation to the associated sentence of the second translation
- Match sentences where the cosine measure applied on embeddings centroids is maximal

# Translation-Matching



# Bert

- Bert stands for Bidirectional Encoder Representations from Transformers
- Ordinary word embeddings
  - do not consider word senses
  - word context is considered only during training
  - embeddings of a word independent from current word context
- This is different for Bert, Bert embeddings represent the sense (reading) of the current word
- Bert embeddings cannot be stored in global hashtable, but have to be generated on the fly by deep neural network

We used two ways to generate Bert embedding centroids

- average over all Bert word embeddings
- use only embeddings of the “sentence beginning token” representing the entire sentence

# Customer Segmenation

| Method | Accuracy | | | |
|---|---|---|---|---|
| | Contest 1 | Contest 2 | Contest 3 | Total |
| Word2Vec | 0.347 | 0.328 | 0.227 | 0.330 |
| Bert (AW) | 0.046 | 0.223 | 0.061 | 0.118 |
| Bert (ST) | 0.109 | 0.149 | 0.136 | 0.07 |

# Translation Matching

| Method | Accuracy |
|---|---|
| W2VC | 0.726 |
| Bert (ST) | 0.423 |
| Bert (AW) | 0.279 |
| Random | 0.010 |

# Discussion

- Very poor results for Bert embeddings for customer segmentation
- Results for translation matching were better but still lagging behind Word2Vec embeddings
- Possible reasons
  - Use of multilingual model
  - keyword list is arbitrarily ordered and not like ordinary sentence

# Conclusion

- We applied Bert and Word2Vec embeddings on two NLP scenarios
  - Customer segmentation
  - Translation matching
- Bert embeddings clearly outperformed by Word2Vec embeddings
- Next steps: Using
  - other embeddings methods like ELMo
  - monlingual embeddings