CentraleSupélec

BERTScore & MoverScore

Presented to:
Pierre Colombo

Presented by:
Rebecca Bayssari
Paul-Ambroise Leroy
Marius Nadalin

Outline

1. Introduction: Context & State of the art
2. New scoring methods: BERTScore & MoverScore
3. Experimentations

Introduction

Context & State of the art

The Papers

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger.
MoverScore: *Text generation evaluating with contextualized embeddings and earth mover distance.*
In EMNLP, **2019**.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi.
BERTScore: *Evaluating text generation with BERT.*
In CoRR, **2019** (Published in ICLR, 2020)

Objective: Automatic evaluation of natural language generation

Reference text	People like foreign cars.
Candidate text 1	People like visiting places abroad.
Candidate text 2	Consumers prefer imported cars.

Score 1

Score 2

Paper Motivations

State of the art: a problematic example

Reference text	People like foreign cars.		
Candidate text 1	People like visiting places abroad.	Score 1	BLEU
Candidate text 2	Consumers prefer imported cars.	Score 2	Human eval

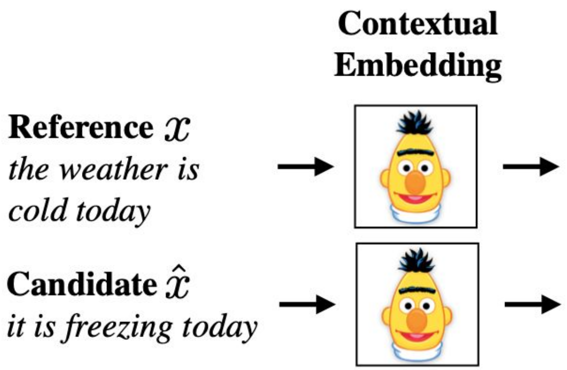
An ideal scoring method would:

- Take **semantics** into account and recognize:
  - **Meaning-preserving vocabulary** (synonyms / paraphrases)
  - **Compositional diversity** (reordering words)
  - **Context**
- With an **unsupervised** training
- Go toward **human-like evaluation** (test on specific tasks with human-labeled data)

New scoring methods

BERTScore & MoverScore

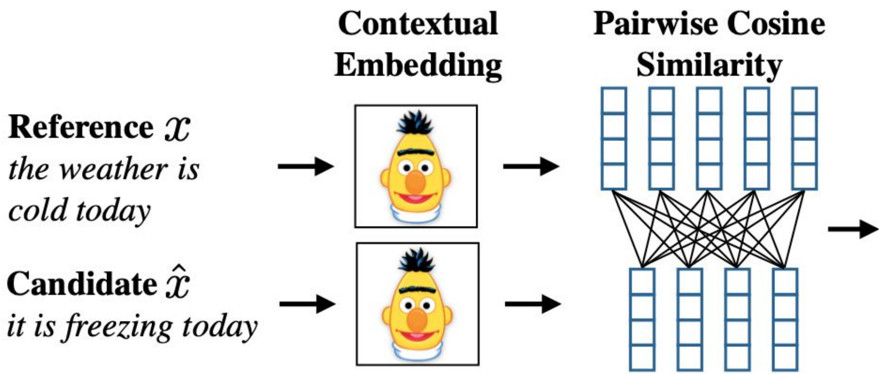
How BERTScore works



Step 1: Tokenization & Embedding

- Use a **pre-trained BERT** model to obtain **contextualized** word embeddings for each token.
- Each token is represented as a high-dimensional **vector**, capturing **semantic meaning**.

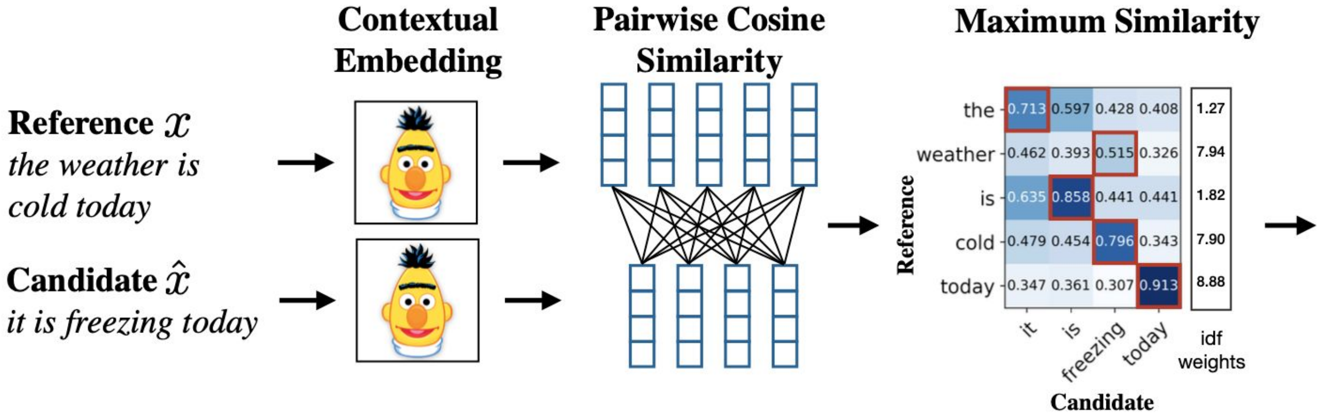
How BERTScore works



Step 2: Pairwise Cosine Similarity

- Compute cosine similarity between embeddings of candidate and reference tokens.
- Measures semantic closeness based on context.

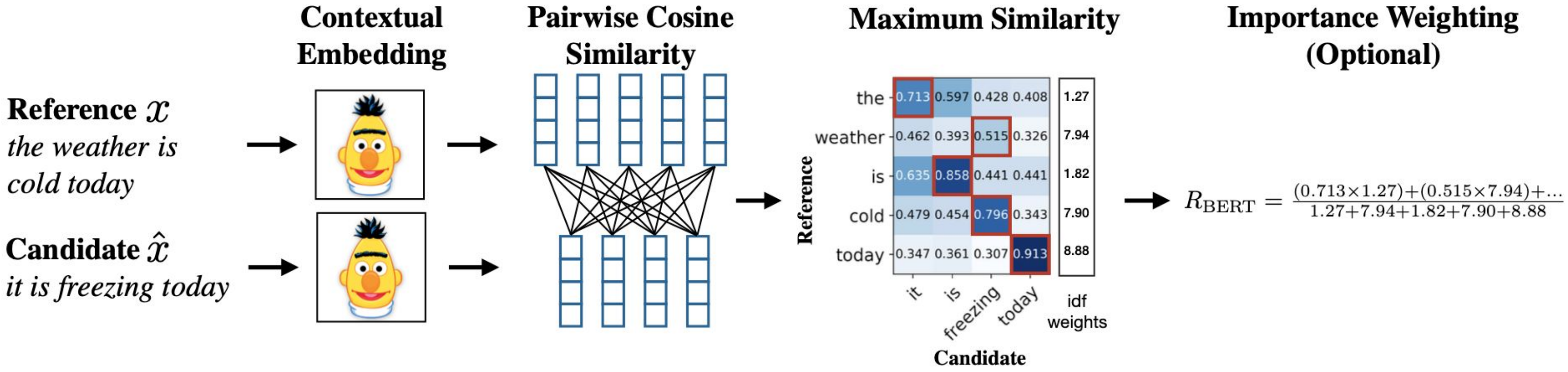
How BERTScore works



Step 3: Matching tokens

- For each token in candidate, **find the most similar token** in reference.

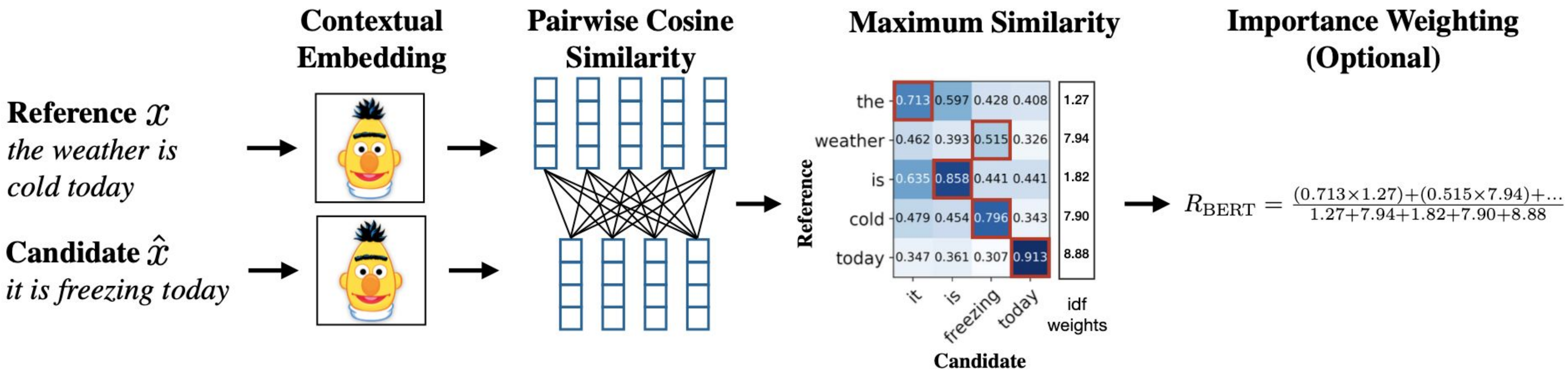
How BERTScore works



Step 4: Precision, Recall and F1 computation

- **Precision**: Average similarity of candidate tokens matched to reference.
- **Recall**: Average similarity of reference tokens matched to candidate.
- **F1 score**: Harmonic mean of precision and recall.

How BERTScore works



Step 5: Weight with IDF (optional)

- Weight the computation of Precision, Recall and F1 score.
- IDF (Inverse Document Frequency) **weights rare tokens more**.

MoverScore

system text x reference y

1-gram (or 2-gram) sequence

$\mathrm{x} = (\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_n)$ $\mathrm{y} = (\mathrm{y}_1, \ldots, \mathrm{y}_m)$

contextual vector representation
(*embedding, can be pre-trained*)

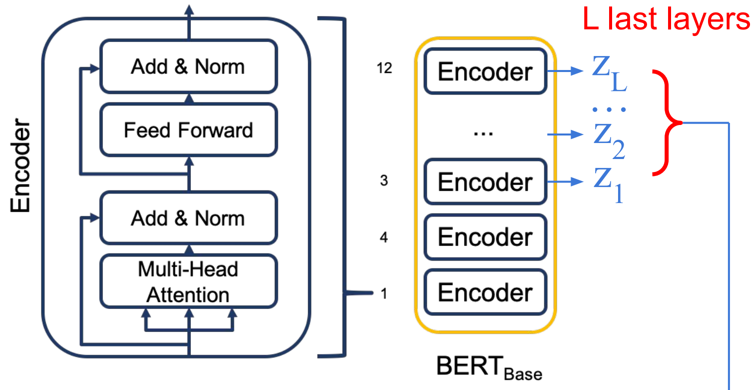
$\mathrm{E}(\mathrm{x}_i)$ $\mathrm{E}(\mathrm{y}_i)$

semantic distance and
transportation matrix **C**

$$C_{ij} = d(\mathrm{x}_i^n, \mathrm{y}_j^n) = ||E(\mathrm{x}_i^n) - E(\mathrm{y}_j^n)||_2$$

$\mathrm{x} = (\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_n)$

embedding with BERT
(trained on MultiNLI)



power means

$$h^{(p)} = \frac{1}{L} \sum_{i=1}^{L} z_i$$
$$h^{(+\infty)} = \max(\{z_i, i \in [[1, L]]\})$$
$$h^{(-\infty)} = \min(\{z_i, i \in [[1, L]]\})$$

concatenate

$$E(x_i) = h^{(p)} \bigoplus h^{(+\infty)} \bigoplus h^{(-\infty)}$$

MoverScore

system text x reference y

1-gram (or 2-gram) sequence

$x = (x_1, x_2, \ldots, x_n)$ $y = (y_1, \ldots, y_m)$

contextual vector representation
(*embedding, can be pre-trained*)

$E(x_i)$ $E(y_i)$

semantic distance and
transportation matrix **C**

$$C_{ij} = d(\mathrm{x}_i^n, \mathrm{y}_j^n) = ||E(\mathrm{x}_i^n) - E(\mathrm{y}_j^n)||_2$$

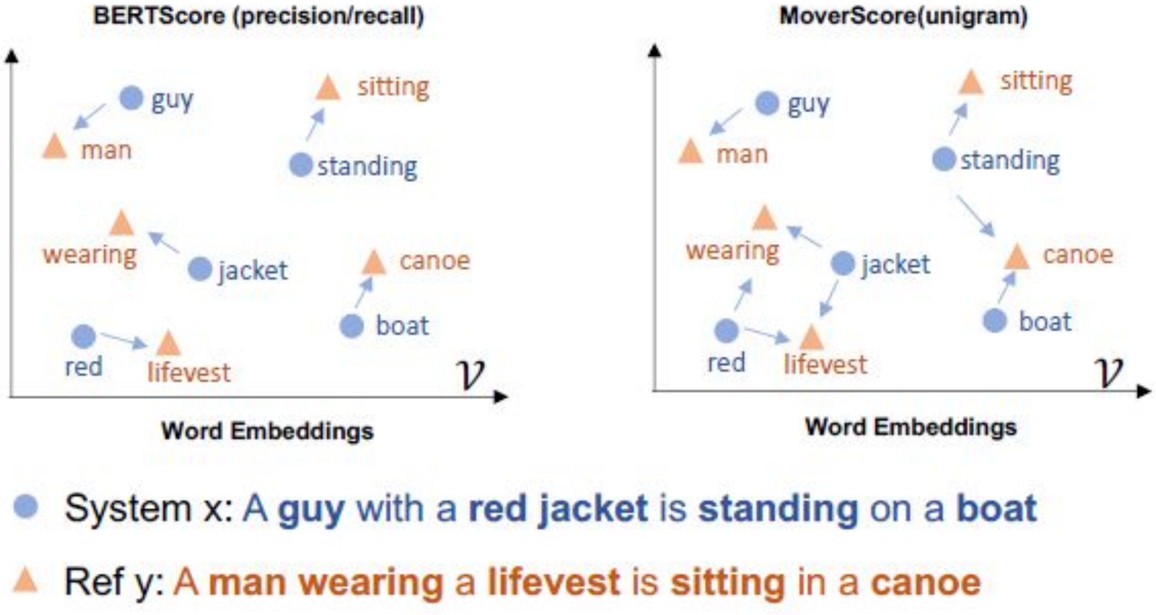
$$\boxed{\begin{aligned}&\mathrm{WMD}(\boldsymbol{x}^n, \boldsymbol{y}^n) := \min_{\boldsymbol{F} \in \mathbb{R}^{|\boldsymbol{x}^n| \times |\boldsymbol{y}^n|}} \langle \boldsymbol{C}, \boldsymbol{F} \rangle, \\ &\text{s.t. } \boldsymbol{F}\mathbf{1} = \boldsymbol{f}_{\boldsymbol{x}^n}, \quad \boldsymbol{F}^\intercal \mathbf{1} = \boldsymbol{f}_{\boldsymbol{y}^n}.\end{aligned}}$$

Where:

$$\boldsymbol{f}_{\boldsymbol{x}_i^n} = \frac{1}{Z} \sum_{k=i}^{i+n-1} \mathrm{idf}(\mathrm{x}_k)$$ is a distribution of weights

→ **BERTScore** (precision/recall) can be represented as a (non-optimized) **Mover Distance**

Interpretation



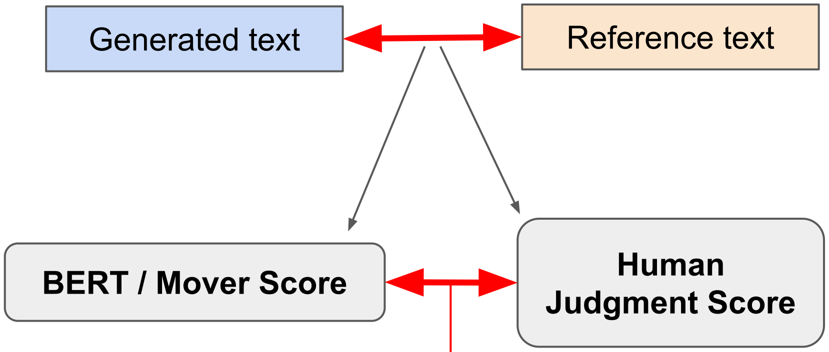
An illustration of MoverScore (hard alignments) and BERTScore (soft alignments)

Experimentations

Test human-like performance & Comparison

Experimentation Setup

- Objective: test scoring against human judgment
- Tasks:
  - Machine translation
  - Image captioning
  - Text summarization
  - Data-to-text generation



Pearson correlation coefficient

$$\boxed{\rho_{X,Y} = \frac{\mathrm{cov}(X,Y)}{\sigma_X \sigma_Y}}$$

Results for BERTScore

Metric	cs ⟷ en	de ⟷ en	et ⟷ en	fi ⟷ en	ru ⟷ en	tr ⟷ en	zh ⟷ en
BLEU	.970	.971	.986	.973	.979	.657	.978
ITER	.975	.990	.975	.996	.937	.861	.980
RUSE	.981	.997	.990	.991	.988	.853	.981
YiSi-1	.950	.992	.979	.973	.991	**.958**	.951
P_{BERT}	.980	.998	.990	.995	.982	.791	.981
R_{BERT}	**.998**	.997	.986	.997	**.995**	.054	**.990**
F_{BERT}	.990	**.999**	.990	**.998**	.990	.499	.988
F_{BERT}(idf)	.985	**.999**	**.992**	.992	.991	.826	.989

Pearson Correlation with system-level human judgments on WMT18 (translations to English)

Key takeaways

F1(BERT)
It balances precision and recall, making it a reliable metric for capturing both the accuracy and completeness of a generated text

IDF Weighting
Provides small benefits in certain cases but does not consistently improve performance.

BERTSCORE Robustness: Image Captioning

Dataset

The evaluation uses the **COCO** dataset with five human-written reference captions for each image

Metrics Compared

BERTSCORE is compared against general-purpose metrics like BLEU and caption-specific metrics like SPICE and LEIC.

BERTSCORE Robustness: Handling Adversarial Challenges

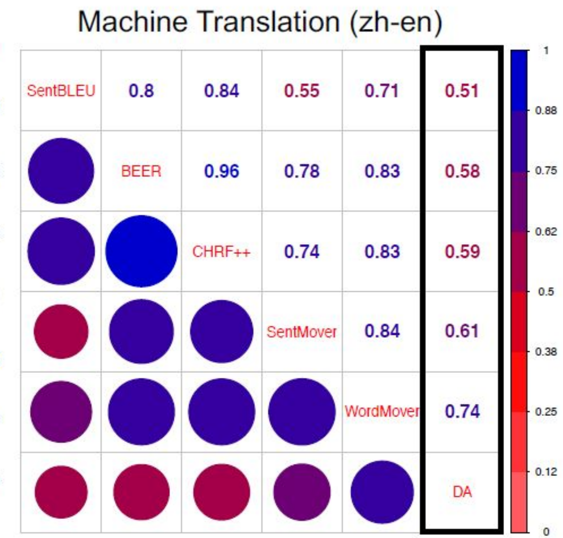
Datasets

QQP: Standard paraphrase detection tasks.
PAWS: Tough adversarial examples with word swaps and reordered phrases.

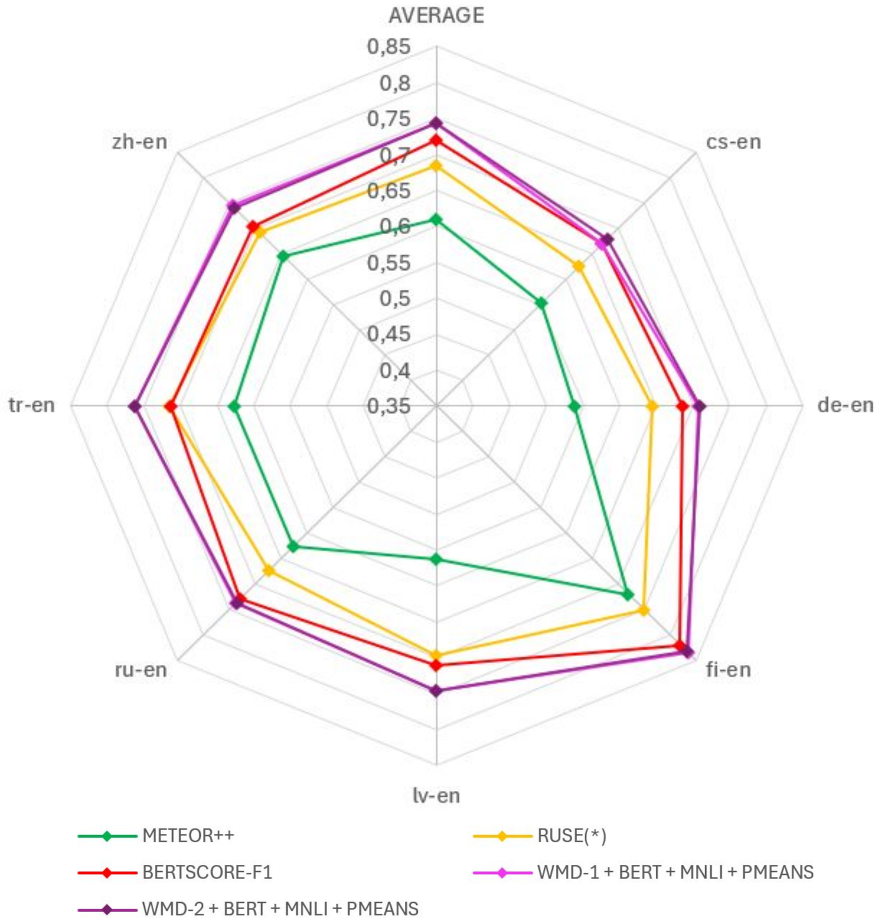
AUC Scores

The Area Under the ROC Curve (AUC) is used to evaluate the performance of metrics and models.

Results for MoverScore



Correlation in distant language (zh-en) pair



Absolute Pearson correlations with segment-level human judgments in 7 language pairs on WMT17 dataset

Conclusion

- The two scoring methods aim to automatically evaluate text-generation based on semantics.
- They both rely on the BERT contextual embeddings.
- Although **BERTScore** surpasses traditional metrics by effectively capturing semantic similarity between candidate and reference captions, **MoverScore** stands out as a more advanced and precise metric, offering superior performance in evaluating text quality.

Appendix

Understanding Contextual Embedding Layers

Dataset

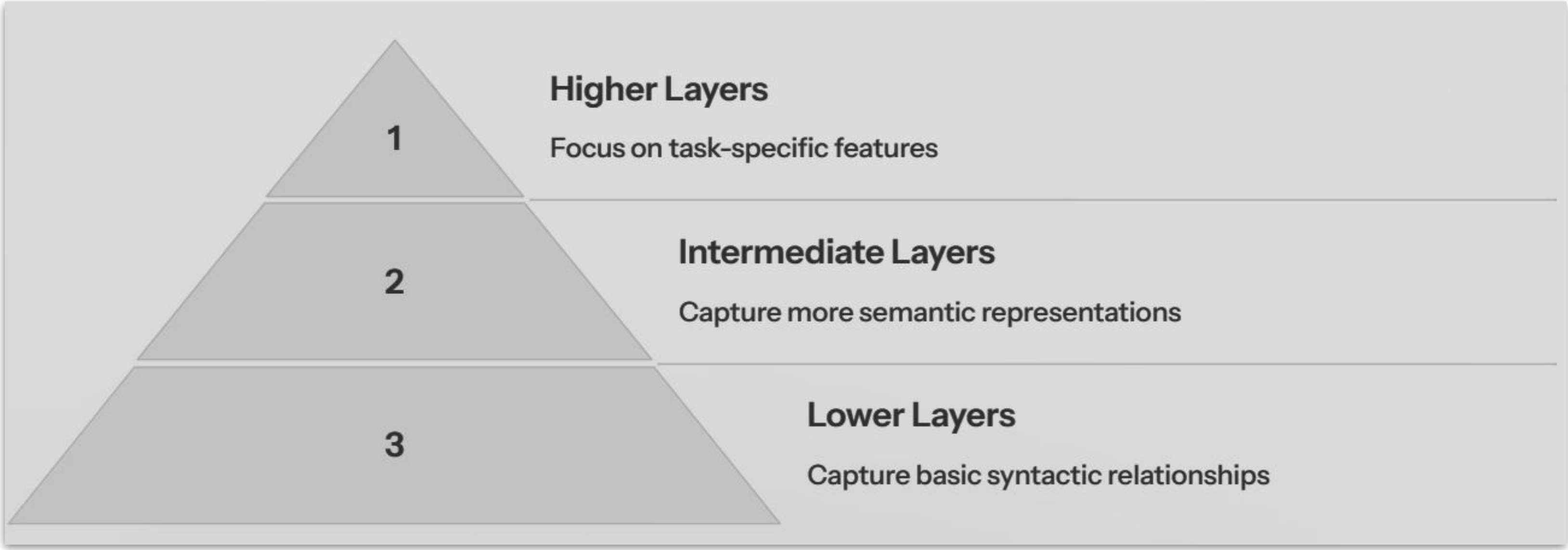
The WMT16 dataset, from the Workshop on Machine Translation 2016, serves as the validation set for this study.

Purpose

It's used to determine which layer from each model produces the best embeddings for semantic tasks.

Approach

A systematic evaluation ensures the optimal layer is selected for each contextual embedding model.



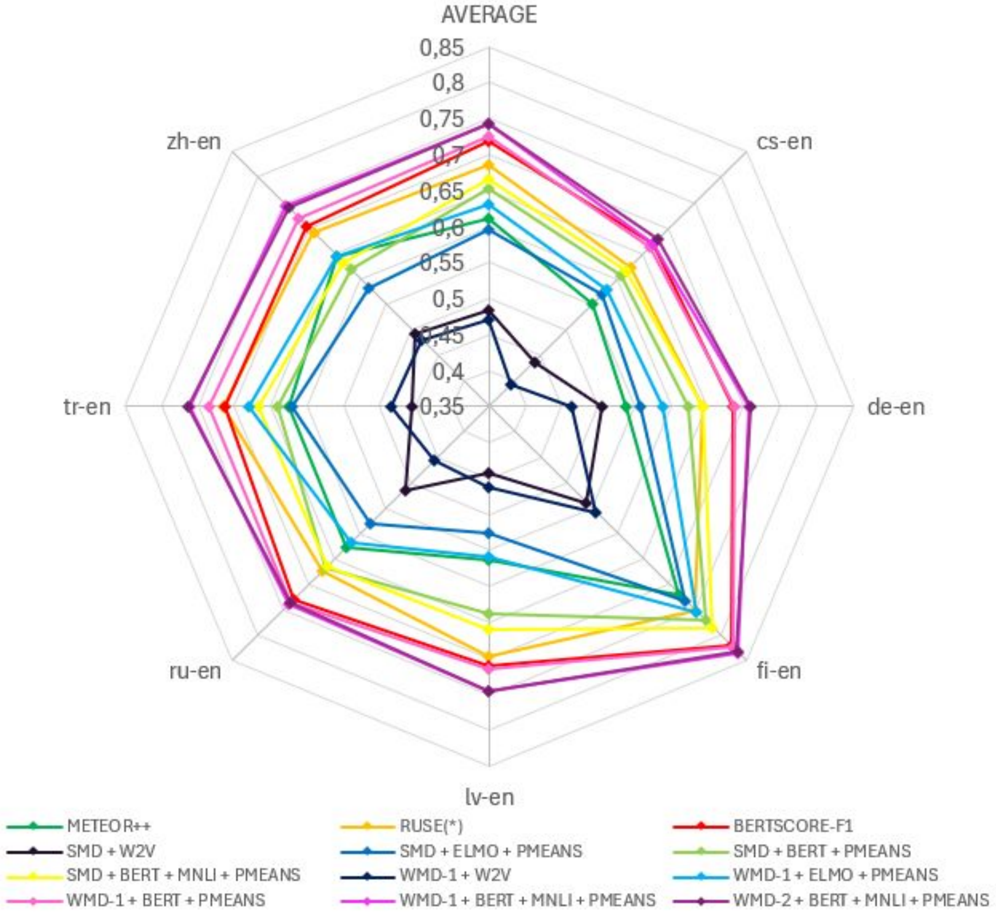
Empirical Results

4 different tasks :

- Machine translation
- Text summarization
- Data-to-text generation
- Image Captioning

Pearson correlation coefficient :

$$\rho_{X,Y} = \frac{\mathrm{cov}(X,Y)}{\sigma_X \sigma_Y}$$



Absolute Pearson correlations with segment-level human judgments in 7 language pairs on WMT17 dataset

Current solutions: BLEU & METEOR

1

n-grams

Candidate text: People like visiting places abroad.

- 1-grams: “People”, “like”, “visiting”, “places”, “abroad”
- 2-grams: “People like”, “like visiting”, “visiting places”, “places abroad”
- 3-grams: “People like visiting”, “like visiting places”, “visiting places abroad”

Reference text: People like foreign cars.

2

Statistics

Precision: $\text{Exact-P}_n = \dfrac{\text{Number of \textbf{candidate} n-grams that \textbf{match reference} n-grams}}{\text{Number of candidate n-grams}}$

Recall: $\text{Exact-R}_n = \dfrac{\text{Number of \textbf{reference} n-grams that \textbf{match candidate} n-grams}}{\text{Number of reference n-grams}}$

3

Score

BLEU ~ Geometric average of Exact-P_n for n=1,2,3,4

METEOR ~ Exact-P1 and Exact-R1 with relaxed matching