CAN LLM AS ENCONDER GOOD?

Task: Entity Matching

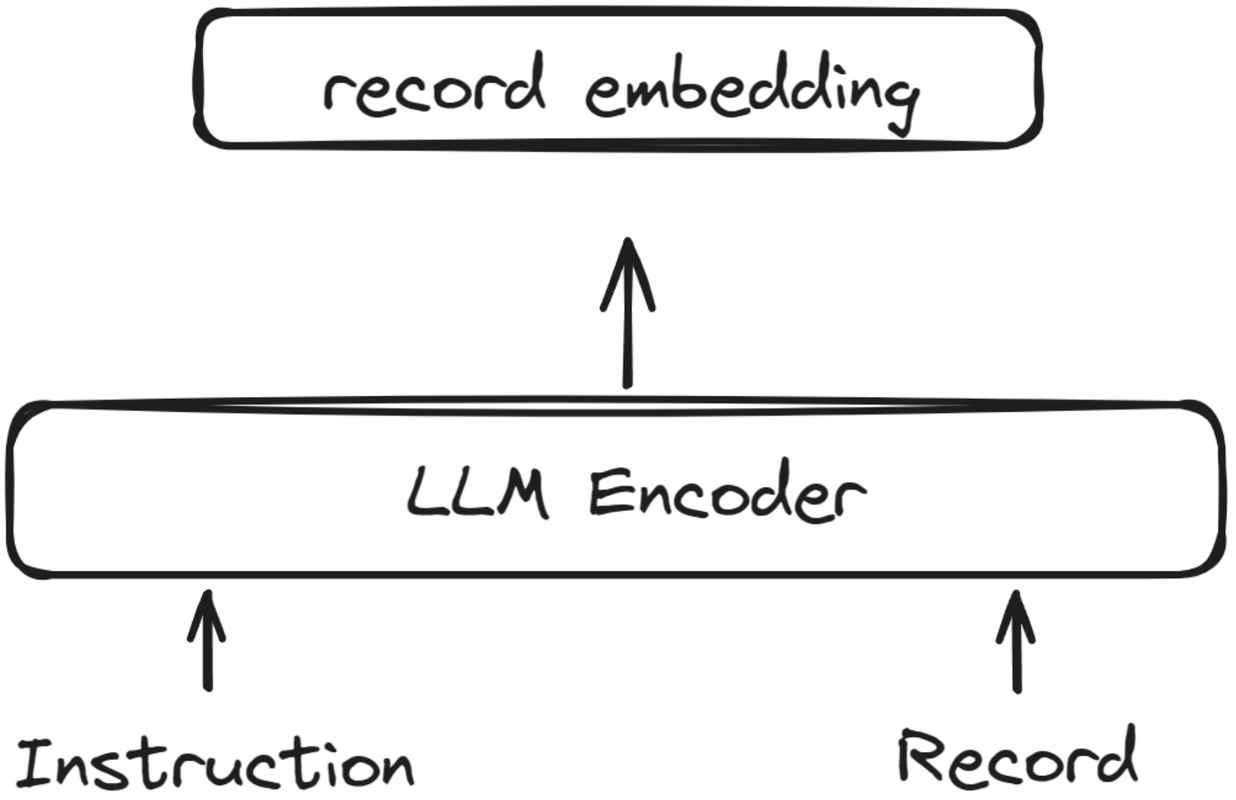
Starting problem

LLM achieve state of the art in the tast of Entity Matching in the majority of the dataset, but have several problem like:

- opacity;
- unable to provide numeric accuracy

Idea

Using the idea of **LLM2Vec: Large Language Models Are Secretly Powerful Text Encoders**, we want to embed a record and use numerical vector instead of generative answer for evaluation.



LLM2Vec

briefly

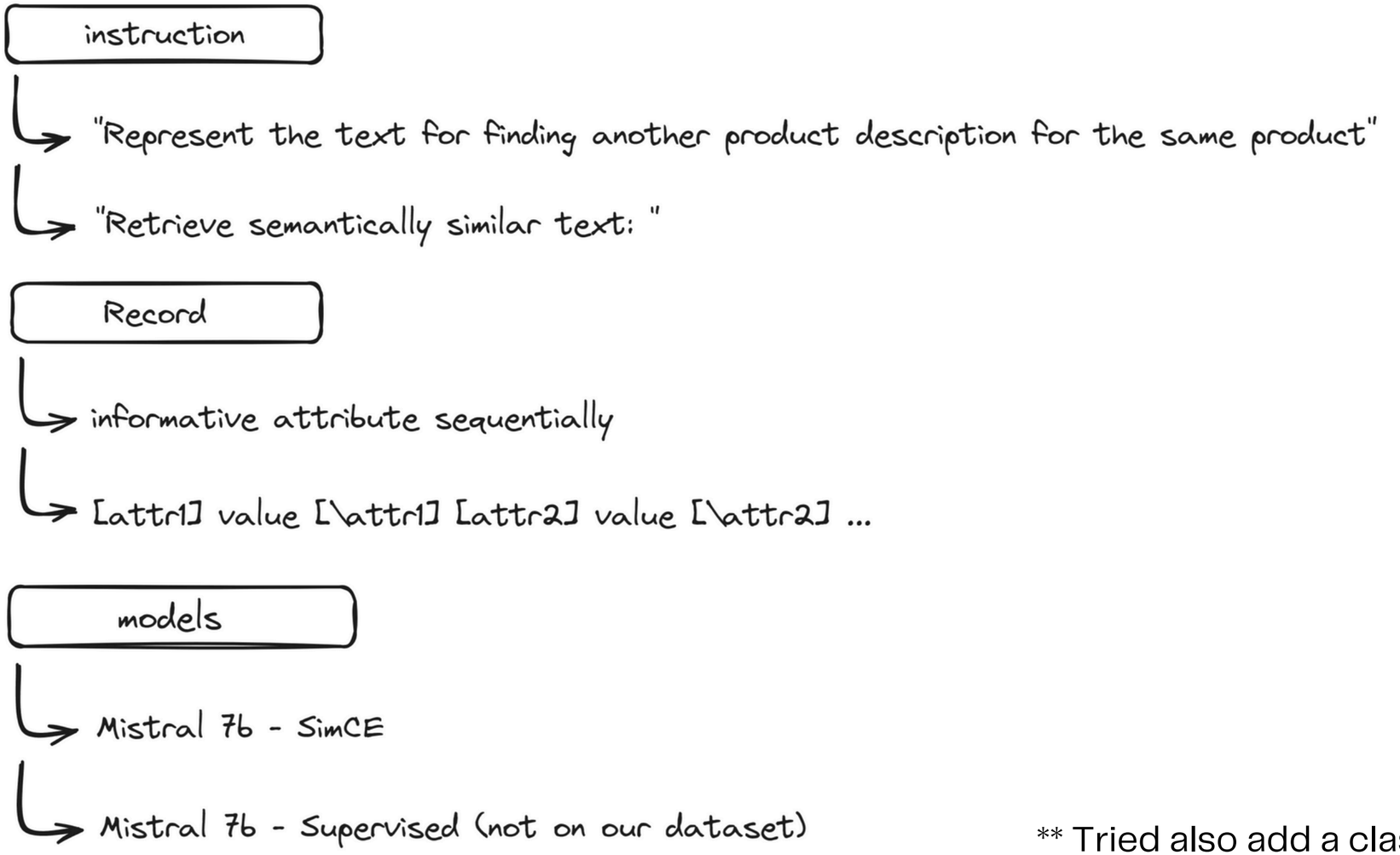
1. **Enabling bidirectional attention (Bi):** The first step of the LLM2Vec approach is to replace the causal attention mask of decoder-only LLMs by an all-ones matrix. This gives each token access to every other token in the sequence, converting it into a bidirectional LLM.
2. **Masked next token prediction (MNTP):** training objective that combines next token prediction with masked language modeling. Used to adapt a model to make use of its bidirectional attention.
3. **Unsupervised contrastive learning (SimCSE):** The model is trained to maximize the similarity between two representations while minimizing the similarity with representations of other sequence in the batch.

Dataset

Standard dataset taken by github Deepmatcher for benchmarking.
The first is clean, the others are dirty by inserting noise in the records.

	A_size	B_size	testing_pairs	num_pos_match	num_neg_match	attributes
abt_buy	1081	1092	1916	206	1710	3
dirty_dblp_acm	2616	2294	2473	444	2029	4
dirty_dblp_scholar	2616	64263	5742	1070	4672	4
dirty_itunes_amazon	6907	55923	109	27	82	8
dirty_walmart_amazon	2554	22074	2049	193	1856	5

Approches



** Tried also add a classifier on top with null result

Results

Result using Cosine Similarity with a threshold of 0.5 using Mistral 7B Supervised with the records in sequential structure.
Instruction text: "Represent the text for finding another product description for the same product"
Other configuration achieve the same performance.

	abt_buy_exp_data	dirty_dblp_acm_exp_data	dirty_dblp_scholar_exp_data	dirty_itunes_amazon_exp_data	dirty_walmart_amazon_exp_data
precision	0.107516	0.179612	0.186671	0.247706	0.094192
recall	1.000000	1.000000	1.000000	1.000000	1.000000
F1	0.194156	0.304527	0.314613	0.397059	0.172168

The pipeline tend to create a lot of FP (by classifing match).
What is surprising is that achieve better result on dirty dataset.

Conclusion

Following the instruction of the author I try to represent the record using an LLM Encoder.
Despite the SOTA result of the LLM in general, for Entity Resolution porpuse we didn't achieve the same results.

The possible explanation can be:

- the llm doesnt take into account the complete pair, but encode only separate sentences
- the instruction, for this aim, is not perfect for pair classification. The authors try clustering / binary classification and retrieval which is different with our more precise task.

Despite the problem, using this tecnique does infact provide a numerical and more precise accuracy in order to evaluate an LLM, but the embedding time is still huge with respect simpler models.