UNIVERSITÀ DEGLI STUDI DI MILANO
FACOLTÀ DI SCIENZE E TECNOLOGIE

Corso di Laurea Magistrale in Informatica (LM-18)

Natual Language Processing project:

Hurry up, I'm hungry!

Professor: Dott. Alfio Ferrara

Student: Andrea Moretti, matricola: 47263A

Year 2023/2024

1 Introduction

The project aims at developing a system that generates culinary recipes based on a given list of ingredients. The preparation of food is a complex and subjective process that makes it challenging to determine under-lying rules for automation.

The system recives a sentence in input containing the list of the ingredient you want to use and then, using a large language model (LLM), it will compose ingredients and cooking methods to produce a sentence with all the preparation steps of a recipe. To create a model for this purpose I adopted NLP and Deep Learning techniques.

The main objective for achieve this behavoir is to use an existing model with seq2seq capabilities and enhance its understanding of culinary contexts. Therefore, the model is finetuned with a recipes dataset to learn how to produce a sentence using the previous knowledge.

At the end of this project the system will be evaluated using consistent metrics and comparing the similarity between the recipe produced and the expected one, in order to objectively measure the system's performance.

2 Research question and methodology

To reach good performances it's important to explore all the possible existing way for the creation of a model with this capabilities. One possible approch is proposed by researchers of University of Bonn: extracting the data from existing recipes using natural language processing, to learn the combination of ingredients, preparation actions and cooking instructions, and autonomously generates the recipes. *AutoChef*, their model, uses Genetic Programming to represent and evolve the recipes. The fitness of recipes is designed to evaluate the combination of ingredients, actions and cooking-processes learned from the existing recipe data [1].

However, for this project I decided to try a different approach with the aim to explore the power of transformers in NLP task and eventually find their limitations. In the recent years, transformers turned out to be the most efficient structures in the field of NLP; for this reason I decided to use a transformer model as basis for a specialized model in the culinary field.

The project was developed according to the following points:

1. Search for a performant model in seq2seq task;
2. Enhance its knowledge in the culinary context;
3. Search for a dataset composed of examples of recipes with the relative preparation steps and ratings provided by users;
4. Finetune the model on the datasets;
5. Evaluate the performances.

At first I searched for a model with very good performance on seq2seq task. A research of Peking University shows the performances of some famous seq2seq free access models tested on 4 different datasets.

	E2E	WikiBio	WebNLG	ToTTo	Average
Transformer	76.88	81.31	76.32	45.41	69.98
Pointer-GEN	86.97	82.98	78.76	54.57	75.82
T5-small	86.04	86.28	93.92	85.44	87.92
T5-base	**96.36**	**91.38**	**94.10**	88.59	**92.61**
BART-base	91.55	86.37	93.43	**90.71**	90.52
Average	87.56	85.66	87.31	72.94	

Table 3: Human evaluation scores of each model on each dataset (higher means better).

Figure 1

The performances were evaluated by human and also using BLEU score getting very similar result and, as we can see in Figure 1, the best performances were obteined by T5-base [2, 3].

Unfortunately, there is one limitation with this model: it contains a very large number of parameters and with my hardware availability I couldn't train the model efficiently. To face this problem I decided to use the small version of T5 (the T5-small).

Once the model has been chosen, the idea is to enhance its culinary context using the NLP task called Masked Language Modeling (MLM). I searched online for a dataset containing recipes, with preparation steps and user ratings; on Kaggle I found a dataset with just over 200k rows containing crawled data from *Food.com*, a website with recipes from all over the world.

I selected almost 150k rows of sentences contained in this dataset cleaned of noisy words and typos, then I trained the T5-small with MLM for 4 epochs. At this point the model was evaluated using ROUGE and BLEU score metrics.

The next step is to train this specialized model in the generation of recipes using the previous dataset. Also in this case, I celaned the dataset from garbage datas that can have a bad impact on the training performance; finally I obteined almost 110k rows composed by the couple [ingredients, preparation steps]. Afterwards, I combined this dataset with the one containing the user ratings, associating for every recipe the average of the ratings.

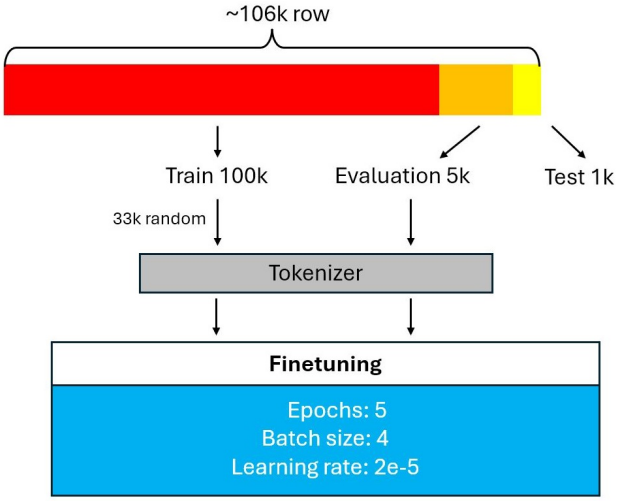

Figure 2

The idea is to use the ratings to weigh the loss during the finetuning: I normalized the rating score so that the model learn more from the most liked recipes.

The training process lasted for 5 epochs, each one used a batch size of 4 and 33k rows randomly chosen from the 100k intended for the train set (Figure 2). At the end of the finetuning the model is able to produce a sentence containing all the preparation steps for a recipe created with the ingredient received as input.

The final step of this project was to evaluate the model with the properly metrics. In order to get objective measures of the model's performances I created a specific metric that takes into account the use of verbs appropriate to the ingredients, producing a normalized score. Furthermore, I used a transformer model (a Sentence-BERT [4]) for sentence embedding to check the similarity between the model generation and the expected sentence.

3 Experimental results

Once the model is trained it's necessary to evaluate its performances; for that I started manually viewing the outputs produced by the model after receving in input some examples from the test set. One first problem I noticed is the start phrase that the model almost always produce: *"preheat oven to 350f"*. This is produced in almost every generated recipes and only in very few situation it's consistent to the context. This problem is probably caused by the presence of too many phareses with this sentence in the preparation steps of recipes valued with high ratings.

At first I evaluated the base model (T5-small) specialized in culinary context trained with MLM and I obteined the following results:

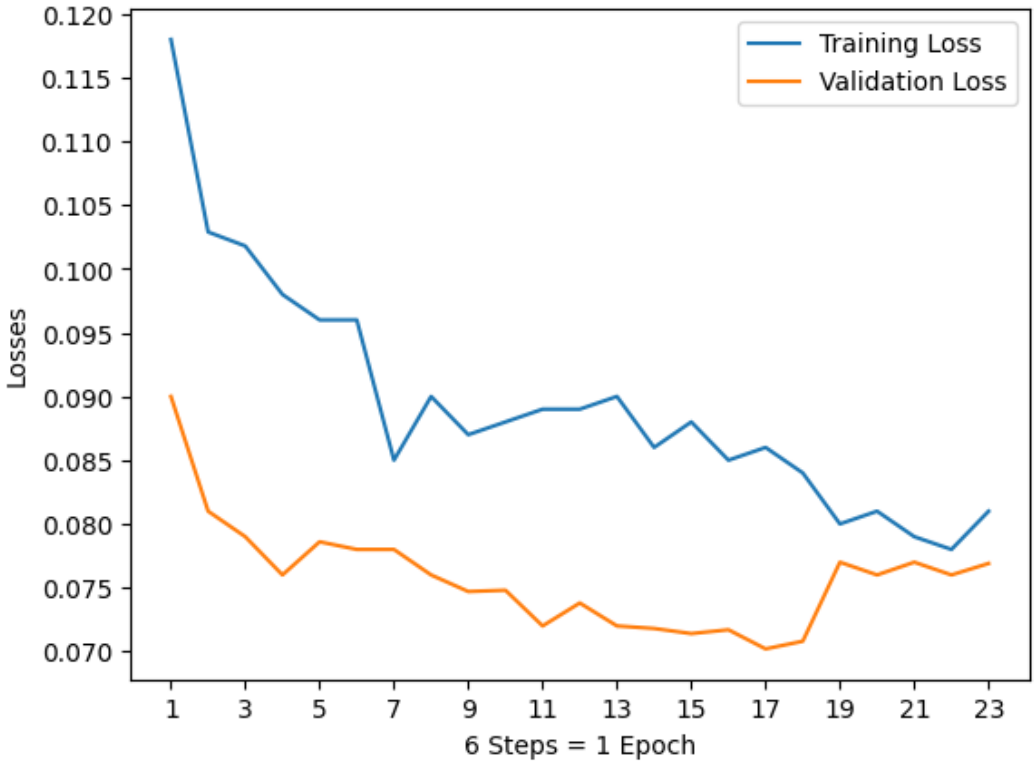

Figure 3

The metric of ROUGE scores at the end of the MLM training:

- rouge1: 0.9259
- rouge2: 0.8627
- rougeL: 0.9258

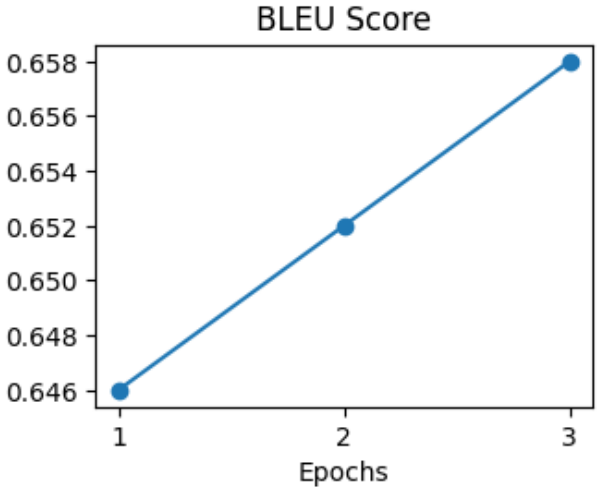

Figure 4

Then I evauated the final model's performances with the metrics described in the previous chapter and I obteined the following results:

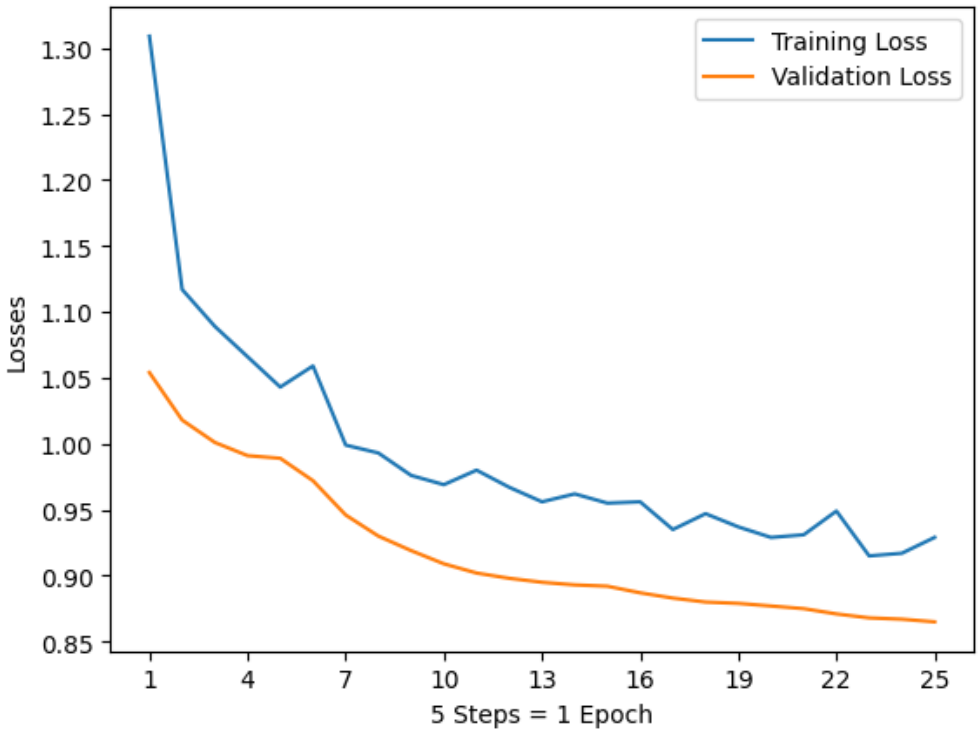

Figure 5

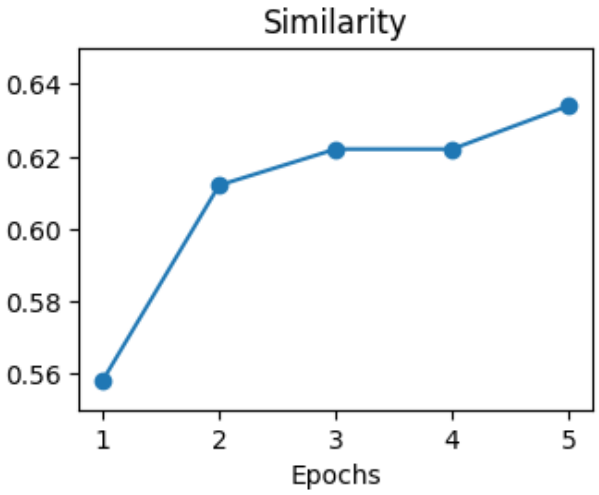

Figure 6

Then I used a metric to evaluate every sentence generated by the model on the test set and works in this way:
For every generated sentences:

- If the verb is *add* (the most used verbs for every ingredient) the model gain 0.75 point;
- If the verb is contained in the top 10 most used verbs associated to the ingredient gain 1 full point;
- 0 in the other cases.

Then the score is averaged.

The final score of the metric is:

0.8587

If the verbs *add* gives 1 full point:

0.9622

4 Concluding remarks

Analyzing the final performances and some examples of generated recipes I figured that the model can probably reach better results with some improves. The model it's capable to produce the preparation steps of a recipe but in many cases the result it's just a mix of ingredients; this shows that the culinary knowledge can be improved.

For this reason another aspect that can be fixed it's the choise of the base model: I used the small version of T5 due to the hardware limitation, using the T5-base can increase the performance of associating correct verbs with ingredients.

Another important thing using transformer models it's the use of a good and large dataset; in order to remove the noise sentence I've dropped more than an half of the original dataset. This was necessary to have a correct learning but probably the amount of data it's not enought to reach best performances.

Using a larger dataset, combined with a better hardware, can significantly improve the precision of the generated sentences, in fact, during the training, the similarity metric (Figure 6) between the generated recipie and the expected one went from 0.56 to 0.64, suggesting room for improvments.

References

[1] H. Jabeen, J. Weinz, and J. Lehmann, "Autochef: Automated generation of cooking recipes," pp. 1–7, 2020.

[2] X. Yin and X. Wan, "How do Seq2Seq models perform on end-to-end data-to-text generation?," pp. 7701–7710, May 2022.

[3] J. Lee, D. Kim, D. Jung, B. Kim, and K.-W. On, "Exploiting the potential of seq2seq models as robust few-shot learners," 2024.

[4] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," CoRR, vol. abs/1908.10084, 2019.