On Cross-Language Entity Label Projection and Recognition

Paolo Gajo[1,*], Alberto Barrón-Cedeño[1]

[1] *Università di Bologna, Corso della Repubblica, 136, 47121, Forlì, Italy*

Abstract

Most work on named entity recognition (NER) focuses solely on English. Through the use of training data augmentation via machine translation (MT), multilingual NER can become a powerful tool for information extraction in multilingual contexts. In this paper, we augment NER data from culinary recipe ingredient lists by means of MT and word alignment (WA), following two approaches: *(i)* translating each entity separately, while taking into account the full context of the list and *(ii)* translating the whole list of ingredients and then aligning entities using three types of WA models: Giza++, Fast Align, and BERT, fine-tuned using a novel entity-shuffling approach. We depart from English data and produce Italian versions via MT, span-annotated with the entities projected from English. Then, we use the data produced by the two approaches to train mono- and multilingual NER BERT models. We test the performance of the WA and NER models on an annotated dataset of ingredient lists, partially out-of-domain compared to the training data. The results show that shuffling entities leads to better BERT aligner models. The higher quality NER data created by these models enables NER models to achieve better results, with multilingual models reaching performances equal to or greater than their monolingual counterparts.

Keywords

information extraction, named entity recognition, cross-lingual label projection, data augmentation

1. Introduction

Named entity recognition (NER) is a sequence labeling task with a long history of works mainly focusing on the recognition of entities such as people, locations, and organizations. Multilingual NER has also attracted research efforts, with recent SemEval campaigns including tasks on multilingual complex NER (MultiCoNER) [1, 2]. Despite its popularity and various mono- and multilingual NER datasets being available, specific domains such as the culinary one likely require new annotated data. In addition, NER is often the first step in information extraction for knowledge graph construction and, to the best of our knowledge, all literature in the domain of cuisine on this topic solely focuses on English data [3, 4, 5, 6, 7]. Therefore we argue that, given cuisine's multicultural nature, more research in this direction is warranted.

Entity label projection [8] aims to address this scarcity by automating the data generation process for NER. This task consists in taking the labels associated with spans from a source text and automatically applying them to its translation in another language, i.e. the target text. Through this task, we attempt to find an efficient automatic way of developing models for entity projection across languages to produce high-quality multilingual data for recipe Named Entities (r-NE) [4]. Departing from an English-language dataset containing ingredients from culinary recipes, annotated at the span level with entity category labels, we first rely on a MT engine to translate each source entity s_i individually into Italian, while keeping the full context into account. This results in a first entity-wise (EW) translated EN–IT–ES dataset where entities are linked across languages.[1]

Using these synthetic alignments, we train BERT models to align source and target entities, shuffling the latter to prevent the model from learning to simply predict the original entity order. We then test the models on two novel entity alignment datasets, partially out-of-domain compared to the training data, e.g., as regards the used food products, units of measure, and cooking processes. As baselines to evaluate the BERT alignment models, we use Giza++ [9] and Fast Align [10], two statistical word alignment (WA) models. In order to produce higher-quality r-NE data, we translate the ingredient lists across their whole length, predicting target entity spans with the best BERT models from the previous step, along with the baseline models. We thus obtain various sentence-wise (SW) translated datasets in Italian, trading some alignment accuracy for better translations.

Both types of training data, EW and SW, are then used to fine-tune mono- and multilingual BERT NER models on the task of recognizing entities in food recipes. The models are trained on various combinations of mono- and multilingual data and are tested on the entity annotations from the two aforementioned novel testing datasets.

Our contribution is three-fold: *(i)* We show the efficacy of fine-tuning alignment models by shuffling entities in contexts where most of the information depends on the presence of lexical items rather than the dependencies

CLiC-it 2024: Tenth Italian Conference on Computational Linguistics, Dec 04 — 06, 2024, Pisa, Italy

[*] Corresponding author.

paolo.gajo2@unibo.it (P. Gajo); a.barron@unibo.it (A. Barrón-Cedeño)

0009-0009-9372-3323 (P. Gajo); 0000-0003-4719-3420 (A. Barrón-Cedeño)

[1] Experiments on Spanish (ES) are included in Appendix A.

linking them. *(ii)* We showcase the performance delta between mono- and multilingual NER models when fine-tuning on the synthetic data produced by our alignment. These models can be used to label large datasets in multiple languages at a finer granularity level compared to currently available monolingual resources. *(iii)* We release code and data to produce data in multiple languages.[2]

The rest of the paper is structured as follows. Section 2 presents relevant past research on the subjects of cross-lingual entity alignment and recognition. Section 3 introduces the datasets and corpora used in the experiments, along with their annotation process. Section 4 presents architecture, training, and evaluation details for the models comprising our pipeline. Section 5 discusses the conducted experiments and their results. Finally, Section 6 summarizes the paper and draws conclusions. Appendix A shows further results including Spanish. Appendix B presents statistics and gives insight on the additional training data used. Appendix C lists information on the computational requirements.

2. Related Work

Word alignment was first approached for statistical MT, with models such as IBM 1-5 [11], used in well-known implementations such as Giza++ and Fast Align. With the advent of Transformers [12] and the BERT model [13], this task has been approached by employing both question answering [14] and token classification [15] models, trained on freely available resources, such as XL-WA [16].

A number of past works have studied label projection following a range of approaches. Jain et al. [8] project `PER`, `ORG`, `LOC` and `MISC` labels (person, organization, location, and miscellaneous) by translating sentences and then finding potential matches using glossaries. Fei et al. [17] align words using Fast Align and use POS tagging to enhance data for semantic role labeling. García-Ferrero et al. [18] use the AWESoME word alignment model [19] to align machine-translated data from NER datasets in seven languages. Li et al. [15] fine-tune a NER model on English `PER`, `ORG`, `LOC`, `MISC` data from CoNLL2003 [20] to infer on the source portion of parallel Opus corpora [21] with the aim of creating silver NER data. Subsequently, they train an XLM RoBERTa alignment model by using Wikipedia articles and project the labels on the target portion of the parallel corpus, which they use to train a target-language NER model.

NER can also be approached with large language models (LLM) [22, 23, 24] by prompting them to extract entities from a given text. For example, PromptNER [25] uses chain of thought [26] along with a list of entity definitions to prompt a variety of LLMs, obtaining results on par with SOTA supervised NER systems. Similarly, [27] use in-context learning [28] to evaluate GPT-3 [22] for NER on the CoNLL2003 [20] and OntoNotes5.0 [29] datasets by using retrieval-augmented generation [30] and comparing the results to BERT and models based on graph neural networks [31].

With regard to data specific to the culinary domain, many English-language resources exist in various forms. RecipeDB [32] is an ontology comprising $118\,k$ web recipes which can be used to relate foods and cooking processes to taste profiles and health data. FoodOn [33] is a "farm-to-fork" ontology which provides a structure of relationships between food products across the whole industrial supply chain. Bridging the gap between ontologies and NER datasets, FoodKG [34] is a knowledge graph which can be used to find ingredient substitutions based on dietary health requirements. It is built by leveraging FoodOn and Recipe1M+ [35], a dataset originally intended for learning joint text/image embeddings on over $1\,M$ culinary recipes. Expanding on Recipe1M+, Bień et al. [36] construct RecipeNLG, comprising more than $2\,M$ recipes. It is the biggest food NER dataset to date, but its granularity stops at the sole food product names. More fine-grained silver labels are obtained by Komariah et al. [37], who propose a new methodology to extract entities from AllRecipes.[3] Doing so, they construct FINER, a dataset comprising $64\,k$ recipes with labels predicted by what the authors refer to as a "semi-supervised multi-model prediction technique." The dataset also contains recipe tags such as `vegetarian` and `vegan`, which can be useful for training recipe classifiers. Leveraging RecipeDB [32], a large-scale structured corpus of recipes, [38] generate a synthetic dataset of augmented ingredient phrases and compare the NER performance of various rule-based and neural models.

Despite the wide availability of English-language resources in the culinary domain, other languages are largely understudied. To the best of our knowledge, the only study to approach this domain in a multilingual setting was conducted by Radu et al. [39], who obtain NER tags automatically in English, German, and French by using a regex-based tagger. Our work aims to partially address this gap in past research by focusing on Italian.

3. Data

The entity alignment data used for training is generated through MT starting from TASTEset [40], a dataset comprising ingredient lists from 700 food recipes, annotated at the span level. We use TASTEset because it is human-curated and its annotations are fine-grained. We translate each entity one by one with DeepL,[4] concurrently feeding the whole ingredient list and the single

[2] Resources available at https://github.com/paolo-gajo/food

[3] https://www.allrecipes.com

[4] https://www.deepl.com/en/docs-api

entity as two separate inputs. This provides DeepL with context, improving translation quality and retaining the start and end span indexes in the target text by simply concatenating each translated entity. To the best of our knowledge, DeepL is currently the only MT engine capable of contextually translating a substring taken from a sentence, which is why we are using it in this study. Doing this, we obtain an Entity-wise Machine-translated TASTEset (EMT). Since entities are automatically paired to the source label, the distribution across English and Italian is identical (Table 1).

We also generate shuffled variations of EMT, where the entities within a single ingredient have a probability $p \in \{0.1, 0.2, \ldots, 1.0\}$ of being shuffled, for a total of ten variations. Figure 1 shows an example where entities have been shuffled in the first and third target ingredients. The rationale behind this approach is that, when training on EMT, if the dataset were to be left as-is, the model would simply learn to associate a source entity to the target entity in the corresponding position, since entities are simply translated and replaced in EMT.

Overall, we have 22 different variations of EMT, i.e. the original and the 10 shuffled versions for each of the two types of tokenization (mBERT's WordPiece [13] vs mDeBERTa's SentencePiece [41]). The datasets have to be tokenized during the generation of the dataset because token indexes depend on the tokenizer being used when converted from character-level span annotations.

We produce a second kind of synthetic dataset by first translating the ingredient lists as a whole, and then aligning source and target entities by using the BERT, Giza++, and Fast Align models presented in Section 4. We refer to this type of dataset as Sentence-wise Machine-translated TASTEset (SMT). As Table 1 shows, the SMT dataset produced by the BERT model trained on both XL-WA and the shuffled version of EMT contains slightly fewer entities than the source material. This is due to the fact that at times the models produce impossible predictions, e.g. predicting the end of an entity to be before its start.[5] This problem does not exist with Giza++ and Fast Align, since their alignments are word-based. As additional training data for the BERT models, we use the EN–IT portion of XL-WA. Table 9 in Appendix B reports the size of each of the partitions we used.

For testing, we annotated an English–Italian dataset of recipes, obtained from GialloZafferano (where the English recipes are translated from the Italian ones).[6] For the annotation process, we recruited a professional translator who is a native speaker of Italian, with an MA in Specialized Translation in both English and Spanish. Figure 2 shows the instructions given for the first multi-class entity annotation task, which consider the same entities as TASTEset, and the second cross-language entity-linking annotation task, carried out by the same annotator at a later time. The annotation was carried out in Label Studio.[7]

Class	**EMT** (en/it)	**SMT** (it)	**GZ** (en)	**GZ**(it)
food	4,020	4,017	5,958	6,473
qty.	3,780	3,777	10,186	6,564
unit	3,172	3,159	8,148	4,450
process	1,091	1,090	217	265
phys. q.	793	791	1,245	1,547
color	231	231	482	479
taste	126	125	98	72
purpose	94	94	69	126
part	55	55	220	263
total	**13,362**	**13,259**	**26,631**	**20,272**

Table 1
Dataset class distributions. EMT and SMT refer to the entity- and sentence-wise machine-translated TASTEset. GZ refers to our testing dataset.

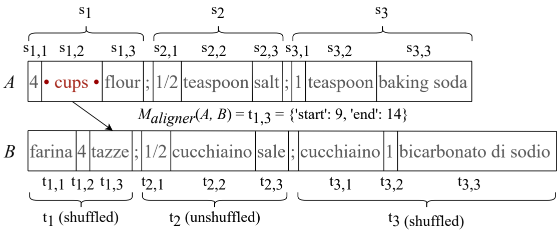

Figure 1: Aligning source s_i and shuffled target t_j entities.

The GialloZafferano (GZ) dataset comprises 597 recipes. The alignments were annotated manually on a subset of 300 recipes, with the possibility of more than one source entity being aligned with one target entity, and vice versa. This is because some recipes contain more than one ingredient option in English but not in Italian (and vice versa), e.g., `Cocomero (anguria) 1 fetta` vs `Watermelon 1 slice`. The GZ dataset contains a total of 46,903 NER annotations and 9,842 alignments.

We manually scrutinized GZ and found that the paired recipes do not always coincide completely. Some ingredients may be missing in either language or be an equivalent rather than the same food product. In order to avoid training the alignment models on excessively different recipes, we chose to avoid annotating alignments whenever the number of source ingredients missing from the target recipe surpassed a heuristic threshold of 1/3.

Note that in GZ quantities and units of measure are localized and are thus listed in both imperial and SI units. As shown in Table 1, this is reflected by the lower number of instances annotated as `quantity` and `unit` in the Italian portion of GZ, compared to its English portion.

[5] The effect on model performance upon training is negligible given that these predictions constitute less than 1% of the total.

[6] https://www.giallozafferano.it

[7] https://labelstud.io

Instructions for the multi-class entity annotation task.

Annotate the ingredients below by assigning to spans of text one of the following categories: `quantity, unit, food, color, part, physical quality, process, purpose, taste`. Use `quantity` for numerical values or expressions such as 'to taste', identifying the quantity of an ingredient. `Unit` stands for "unit of measure", such as grams (g) or ounces (oz). Use `color` for any color that is not part of a `food`'s own name (e.g. 'red' can be tagged in 'red wine', but not 'black' in 'blackberries'). `Part` refers to parts of an ingredient, such as 'wings' in 'chicken wings'. Use `physical quality` for attributes which already characterize an ingredient at the start of the preparation. `Process` refers to actions that the reader is supposed to carry out. The label `purpose` answers the question, "What is this entity used for?" Finally, use `taste` for words referring to a taste, such as 'unsweetened' or 'dry', with relation to a wine. N.B.: **The annotations cannot overlap**. You can either choose to annotate multiple spans with the same label, or avoid annotating one or more spans of text.

Instructions for the cross-lingual entity linking task.

Link each source language entity to its corresponding entity in the target language with an arrow. **Entities should only be linked if they share the same use.** For example, the "2" in "2 tablespoons chopped onions" does not have the same function as in "2 cebollas largas picadas", since one refers to tablespoons and other to the number of onions. However, "onions" and "chopped" could still be linked, as they are equivalent in the two sequences. Individual source entities can be linked to multiple target entities and vice versa. N.B.: **Entities can still be linked if they differ slightly in form or content** but still clearly perform the same function in the same ingredient context. For example, "340" and "450" could still be linked if they both refer to the quantity of grams of the same source and target food products.

Figure 2: Annotation task instructions.

4. Models

Entity Alignment As baselines, we use two statistical models: Giza++ [9] and Fast Align [10]. Giza++ combines a HMM [42] alignment model and IBM M1-5 [11]. Fast Align is much more lightweight, only leveraging IBM M2. We use two multilingual BERT models as well: mBERT [13] as the baseline multilingual Transformer model and mDeBERTa [43] because of its larger size ($276M$ vs $179M$ param.) and performance. When using the BERT models, we follow Nagata et al. [14] and treat entity alignment as a question-answering task, enclosing the source word to be aligned within rarely used characters, e.g., '•', feeding the model both the source sequence A and the target sequence B at once. Figure 1 exemplifies this, where the model $M_{aligner}$ is trained to predict an entity within a shuffled ingredient's boundaries.

Data	*P*	mBERT	mBERT$_{\mathbf{X}}$
EMT	0.0	35.93±0.79	38.87±0.48
	0.1	43.13±2.51	44.49±1.21
	0.2	42.54±1.37	44.02±3.32
	0.3	42.49±3.64	46.61±1.62
	0.4	42.31±2.58	47.04±4.01
	0.5	41.87±1.93	47.22±1.64
	0.6	**44.84±2.19**	46.89±3.36
	0.7	42.87±3.61	47.36±2.06
	0.8	44.08±1.98	**48.34±2.73**
	0.9	42.87±3.27	47.28±1.49
	1.0	41.65±2.25	45.98±1.97
XL-WA	–		21.04

Data	*P*	mDeBERTa	mDeBERTa$_{\mathbf{X}}$
EMT	0.0	42.17±1.19	46.98±3.77
	0.1	57.00±0.94	58.45±1.37
	0.2	55.03±2.40	57.02±2.43
	0.3	57.09 ± 3.61	60.25±2.35
	0.4	57.26 ± 1.09	59.21±2.59
	0.5	55.97 ± 3.11	58.43±2.53
	0.6	**58.37 ± 2.46**	61.07±2.94
	0.7	57.07 ± 1.58	60.68±3.01
	0.8	57.31 ± 1.20	**62.08±3.74**
	0.9	56.95 ± 2.69	61.05±1.27
	1.0	57.59 ± 1.81	60.87±1.13
XL-WA	–		31.71

Table 2
Exact metric results of the alignment task; averaged out of 5 random runs, besides the XL-WA baseline. Best in bold.

We train the models for up to 3 epochs on each dataset with a batch size of 16. The optimizer's learning rate is set at 3×10^{-4}, while ϵ is 10^{-8}. Each training run, we select the best model based on the Exact metric E [44]:

$$E = \frac{\sum_i^n exact(p_i, g_i)}{\|preds\|} \quad , \tag{1}$$

where $preds$ is a list of predictions and $exact(p_i, g_i)$ is the Kronecker delta:

$$exact(p_i, g_i) = \begin{cases} 1, & \text{if } p_i = g_i, \\ 0, & \text{if } p_i \neq g_i \end{cases} \tag{2}$$

with the predicted and gold strings p_i and g_i having been lowercased and stripped of excess punctuation and spaces. We calculate mean Exact and its standard deviation out of five random runs for each model.

In order to improve the models' ability to align entities, we optionally train them on an intermediary word-alignment task using the EN–IT training and dev sets of XL-WA. In addition, we train mBERT and mDeBERTa solely using said XL-WA partitions in order to test them directly on GZ. This serves as a baseline which will allow us to gauge the positive effects of fine-tuning on EMT.

Class	Fast Align	Giza++	mBERT$_\text{X}$	mDeBERTa$_\text{X}$
Qty.	18.41	35.21	30.09	**54.95**
Unit	**30.94**	15.24	24.81	29.75
Food	61.95	77.01	81.66	**83.49**
Process	15.27	51.91	62.60	**83.21**
Color	33.70	84.81	67.04	**85.93**
Phys. q.	39.00	71.76	61.41	**87.66**
Taste	0.00	27.03	35.14	**75.68**
Purpose	25.64	61.54	**94.87**	89.74
Part	52.48	**63.37**	13.86	14.85
Macro avg.	30.82	54.21	52.38	**67.25**

Table 3
Exact metric results of the alignment task by class on GZ for the best models (trained on IT⊕ES). Best in bold.

Entity Recognition For the NER task, treated as token classification, we once again use mBERT.[8] To test the efficacy of the multilingual approach, we also use the following monolingual models when training and testing on a single language: `bert-base-uncased` (henceforth "BERT$_\text{en}$") for English [13] and `bert-base-italian-uncased` ("BERT$_\text{it}$") [45] for Italian. We forgo mDeBERTa for this task, as the focus is showing a comparison between models of equivalent size and performance. Prior to training, the data is preprocessed and labeled using the BIO annotation scheme [46]. We ignore subword tokens when calculating cross-entropy loss, following established methodology.[9]

We train the models on the EN–IT, EN–ES, EN–IT–ES language subsets of EMT and of the four versions of SMT, produced by mBERT, mDeBERTa, Giza++, and Fast Align. For the BERT models, we use the same hyperparameters used for the alignment task, but with a lower learning rate of 2×10^{-4}. The models are evaluated using the macro F$_1$-measure. Details on the employed computational resources can be found in Appendix C.

5. Results and Discussion

Entity Alignment Table 2 reports the Exact scores for the entity alignment experiment. The entity shuffling approach appears to be very effective for creating data which can make the models better at generalizing. The performance of every single model is greatly enhanced when shuffling ingredients just 10% of the time, with increased shuffling frequency not leading to any significant further improvement. The increase in performance seems to be greater for models which have undergone intermediate training on XL-WA, with mDeBERTa$_\text{X}$ gaining almost 12 points in the Exact metric, when fine-tuned on shuffled data. Unsurprisingly, the larger mDeBERTa performs much better than the smaller mBERT across the board. Although the model obtaining the highest mean performance is obtained at $P = 0.8$, an overlap can be observed between all the confidence intervals for $P \geq 0.1$. However, this is not true when going from $P = 0$ to $P = 0.1$. Consequently, increased shuffling past 10% does not seem to provide a concrete performance gain, which is why we decided to produce SMT by using the BERT trained on the least-shuffled version of EMT.

In and of itself, the intermediary training step on XL-WA provides a slight performance boost when looking at mBERT vs mBERT$_\text{X}$ and mDeBERTa vs mDeBERTa$_\text{X}$. Still, this increase is much smaller compared to the one gained through shuffling. While fine-tuning the models on a general word-alignment task can be beneficial, the target domain is likely too different from the training data for this to produce a large performance boost. This is especially true as regards the structure of the sentences, since the test data is comprised by short lists of entities separated by semicolons, while the training data is a domain-balanced sample of sentences from Wikipedia. An additional performance boost is provided by multilingual fine-tuning, while cross-lingual settings (e.g., fine-tuning on ES and testing on IT) lead to worse outcomes. Table 6 (Appendix A) shows the results.

Table 3 reports the performance of the best overall models on each class. As the results show, the much lighter Giza++ model surpasses mBERT$_\text{X}$, only trailing behind mDeBERTa$_\text{X}$. The poor scores achieved by the two BERT models are largely attributable to their poor scores on the `unit` and `part` classes. We hypothesize that this poor class-specific performance has to do with units of measure often being very short strings. Training mDeBERTa only on the `unit` instances does not improve its performance, with the model scoring a lower 18.08 Exact metric. Inspecting its individual predictions in this single-class scenario, we noticed that the model does learn to always predict two consecutive tokens, but the enclosed token does not match the original text when converted into characters. This is due to two separate issues: *(i)* the model selects the wrong span, e.g., selecting an ingredient such as "carote" (carrots) rather than the unit "g" or *(ii)* the model's prediction is empty when converted to characters. Since mBERT and mDeBERTa both have poor performance on this class while using two different tokenization algorithms (WordPiece vs SentencePiece), the problem may lie in the models' tokenizer's token-to-character conversion method.[10] We plan to shed light on this in the future. As regards the `part` class, the poor performance could be explained by the small

[8] We do not use the larger mDeBERTa model due to the computational cost deriving from the number of language combinations.

[9] https://huggingface.co/docs/transformers/en/tasks/token_classification

[10] https://huggingface.co/docs/transformers/en/main_classes/tokenizer#transformers.BatchEncoding.char_to_token

Train	Test	Aligner	NER	F_1
it	it	–	mBERT	0.89±0.01
		mBERT$_X$		0.91±0.02
		mDeBERTa$_X$		**0.94±0.01**
		Fast Align		0.84±0.01
		Giza++		0.87±0.03
		–	BERT$_{it}$	0.86±0.01
		mBERT$_X$		0.9±0.04
		mDeBERTa$_X$		**0.94±0.0**
		Fast Align		0.85±0.04
		Giza++		0.91±0.03
en	it	–	mBERT	0.79±0.05
	en			0.9±0.01
	en		BERT$_{en}$	0.91±0.01

Table 4
Model performance for the entity recognition task, in terms of F_1 measure. All results are macro avg. out of 5 random runs.

number of training instances (55). However, the models obtain high scores on the `purpose` class, also just 94 instances (mBERT$_X$ gets 94.87 Exact score). Unfortunately, repeating the approach we used for the `unit` class is not feasible, as fine-tuning the model on just 55 instances does not produce any reliable results ($E_{part} = 3.96$), meaning this will have to be left for future work.

The rest of the results from Table 3 are generally in line with the average results from Table 2. The scores achieved by the baselines for each class do not have any evident outliers, save for Fast Align scoring a 0 on `taste`. More generally, Fast Align, being the simplest and most lightweight model, performs on average well below the other more complex models.

Entity Recognition Table 4 reports the results for the NER task. The aligner column indicates which alignment model, out of the best ones listed in Table 3, has produced the SMT training data used to fine-tune the NER model. When no alignment model is specified, the training data being used is EMT. Note that in this case we are not using EMT's shuffled versions, as there is no relation between any two recipes when fine-tuning on the NER task.

When training and testing on Italian data, the best results are obtained for both mBERT and BERT$_{it}$ when fine-tuning on SMT data produced by mDeBERTa. When fine-tuning them on EMT, the performance is noticeably lower, with a 5-point difference for mBERT and an 8-point difference for BERT$_{it}$. The data produced by mBERT also allows both models to outperform the EMT baseline, although by smaller amounts. Conversely, the data produced by Fast Align and Giza++ worsens the data quality in 75% of the cases. When fine-tuning mBERT on bilingual ES-IT data, the performance on the test set remains essentially unvaried (see Table 8 in Appendix A).

Looking at the baselines at the bottom of Table 4, we can see that fine-tuning mBERT on English data yields worse performance when testing on GZ, compared to fine-tuning on EMT's Italian data. Our data augmentation strategy is thus providing an evident performance boost, with entity alignment producing bigger improvements than machine-translating each entity individually.

In all settings, mBERT performs on par with the monolingual models. This shows that a single multilingual model can suffice when extracting entities from multilingual corpora, saving time and compute.

6. Conclusions

We explored a simple novel technique to automatically generate high-quality multilingual NER data by combining machine translation and cross-language entity linking. For our experiments, we relied on the English-language TASTEset dataset, which includes recipes whose lists of ingredients are span-annotated for entity recognition. Moreover, we manually curated a novel English–Italian cross-language dataset, featuring the same kind of annotation, with the addition of cross-language alignments.

We machine translated the entities in TASTEset's recipes individually and shuffled them within ingredient boundaries. Leveraging this augmented data, we then fine-tuned BERT entity-alignment models. Using statistical word-alignment models as baselines, we tested these BERT models on our English–Italian parallel corpus. The results showed that models fine-tuned using our novel approach consistently outperform those trained on unshuffled data, along with two statistical baselines.

We then created additional synthetic data by first translating TASTEset's recipes in their entirety, and then aligning the entities in the machine-translated target text using the best models obtained from the first part of the study. These data allowed us to obtain better NER models, compared to the ones we would have obtained by using the original recipes translated entity by entity. We tested monolingual English and Italian BERT models against mBERT, and showed that the latter is capable of obtaining the same performance as its monolingual counterparts when tested on monolingual NER data.

In future work, we plan to extend the annotation of our datasets, both in terms of number of instances and annotators. We will also prioritize solving the token-to-character conversion issues encountered in this study. Furthermore, we plan to leverage this data augmentation technique in order to improve multilingual text-to-graph models, since all of the literature in this regard focuses on English-only data [3, 4, 5, 6, 7].

References

[1] S. Malmasi, A. Fang, B. Fetahu, S. Kar, O. Rokhlenko, SemEval-2022 task 11: Multilingual complex named

entity recognition (MultiCoNER), in: G. Emerson, N. Schluter, G. Stanovsky, R. Kumar, A. Palmer, N. Schneider, S. Singh, S. Ratan (Eds.), Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), Association for Computational Linguistics, Seattle, United States, 2022, pp. 1412–1437. URL: https://aclanthology.org/2022.semeval-1.196.

[2] B. Fetahu, S. Kar, Z. Chen, O. Rokhlenko, S. Malmasi, SemEval-2023 task 2: Fine-grained multilingual named entity recognition (MultiCoNER 2), in: A. K. Ojha, A. S. Doğruöz, G. Da San Martino, H. Tayyar Madabushi, R. Kumar, E. Sartori (Eds.), Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023), Association for Computational Linguistics, Toronto, Canada, 2023, pp. 2247–2265. URL: https://aclanthology.org/2023.semeval-1.310.

[3] C. Kiddon, G. T. Ponnuraj, L. Zettlemoyer, Y. Choi, Mise en Place: Unsupervised Interpretation of Instructional Recipes, in: L. Màrquez, C. Callison-Burch, J. Su (Eds.), Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Lisbon, Portugal, 2015, pp. 982–992. URL: https://aclanthology.org/D15-1114. doi:10.18653/v1/D15-1114.

[4] Y. Yamakata, S. Mori, J. Carroll, English Recipe Flow Graph Corpus, in: N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis (Eds.), Proceedings of the Twelfth Language Resources and Evaluation Conference, European Language Resources Association, Marseille, France, 2020, pp. 5187–5194. URL: https://aclanthology.org/2020.lrec-1.638.

[5] D. P. Papadopoulos, E. Mora, N. Chepurko, K. W. Huang, F. Ofli, A. Torralba, Learning Program Representations for Food Images and Cooking Recipes, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, New Orleans, LA, USA, 2022, pp. 16538–16548. URL: https://ieeexplore.ieee.org/document/9878478/. doi:10.1109/CVPR52688.2022.01606.

[6] D. J. Bhatt, S. A. Abdollahpouri Hosseini, F. Fancellu, A. Fazly, End-to-end Parsing of Procedural Text into Flow Graphs, in: N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, N. Xue (Eds.), Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), ELRA and ICCL, Torino, Italia, 2024, pp. 5833–5842. URL: https://aclanthology.org/2024.lrec-main.517.

[7] A. Diallo, A. Bikakis, L. Dickens, A. Hunter, R. Miller, Unsupervised Learning of Graph from Recipes, 2024. URL: http://arxiv.org/abs/2401.12088.

[8] A. Jain, B. Paranjape, Z. C. Lipton, Entity projection via machine translation for cross-lingual NER, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 1083–1092. URL: https://aclanthology.org/D19-1100. doi:10.18653/v1/D19-1100.

[9] F. J. Och, H. Ney, A systematic comparison of various statistical alignment models, Computational Linguistics 29 (2003) 19–51.

[10] C. Dyer, V. Chahuneau, N. A. Smith, A Simple, Fast, and Effective Reparameterization of IBM Model 2, in: L. Vanderwende, H. Daumé III, K. Kirchhoff (Eds.), Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Atlanta, Georgia, 2013, pp. 644–648. URL: https://aclanthology.org/N13-1073.

[11] P. F. Brown, S. A. Della Pietra, V. J. Della Pietra, R. L. Mercer, The mathematics of statistical machine translation: Parameter estimation, Computational Linguistics 19 (1993) 263–311. URL: https://aclanthology.org/J93-2003.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is All you Need, in: Advances in Neural Information Processing Systems, volume 30, Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

[13] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: J. Burstein, C. Doran, T. Solorio (Eds.), Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423. doi:10.18653/v1/N19-1423.

[14] M. Nagata, K. Chousa, M. Nishino, A supervised word alignment method based on cross-language span prediction using multilingual BERT, in: B. Webber, T. Cohn, Y. He, Y. Liu (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 555–565.

URL: https://aclanthology.org/2020.emnlp-main.41. doi:10.18653/v1/2020.emnlp-main.41.

[15] B. Li, Y. He, W. Xu, Cross-Lingual Named Entity Recognition Using Parallel Corpus: A New Approach Using XLM-RoBERTa Alignment, 2021. URL: http://arxiv.org/abs/2101.11112.

[16] F. Martelli, A. S. Bejgu, C. Campagnano, J. Čibej, R. Costa, A. Gantar, J. Kallas, S. Koeva, K. Koppel, S. Krek, M. Langemets, V. Lipp, S. Nimb, S. Olsen, B. S. Pedersen, V. Quochi, A. Salgado, L. Simon, C. Tiberius, R.-J. Ureña-Ruiz, R. Navigli, XL-WA: a Gold Evaluation Benchmark for Word Alignment in 14 Language Pairs, in: F. Boschetti, N. N. Gianluca E. Lebani, Bernardo Magnini (Eds.), Proceedings of the Ninth Italian Conference on Computational Linguistics (CLiC-it 2023), volume 3596, CEUR-WS, Venice, Italy, 2023.

[17] H. Fei, M. Zhang, D. Ji, Cross-lingual semantic role labeling with high-quality translated training corpus, in: D. Jurafsky, J. Chai, N. Schluter, J. Tetreault (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 7014–7026. URL: https://aclanthology.org/2020.acl-main.627. doi:10.18653/v1/2020.acl-main.627.

[18] I. García-Ferrero, R. Agerri, G. Rigau, Model and data transfer for cross-lingual sequence labelling in zero-resource settings, in: Y. Goldberg, Z. Kozareva, Y. Zhang (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2022, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2022, pp. 6403–6416. URL: https://aclanthology.org/2022.findings-emnlp.478. doi:10.18653/v1/2022.findings-emnlp.478.

[19] Z.-Y. Dou, G. Neubig, Word alignment by fine-tuning embeddings on parallel corpora, in: P. Merlo, J. Tiedemann, R. Tsarfaty (Eds.), Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, Association for Computational Linguistics, Online, 2021, pp. 2112–2128. URL: https://aclanthology.org/2021.eacl-main.181. doi:10.18653/v1/2021.eacl-main.181.

[20] E. F. Tjong Kim Sang, F. De Meulder, Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition, in: Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, 2003, pp. 142–147. URL: https://aclanthology.org/W03-0419.

[21] J. Tiedemann, Parallel data, tools and interfaces in OPUS, in: N. Calzolari, K. Choukri, T. Declerck, M. U. Doğan, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, S. Piperidis (Eds.), Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), European Language Resources Association (ELRA), Istanbul, Turkey, 2012, pp. 2214–2218. URL: http://www.lrec-conf.org/proceedings/lrec2012/pdf/463_Paper.pdf.

[22] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language Models are Few-Shot Learners, 2020. URL: http://arxiv.org/abs/2005.14165, arXiv:2005.14165 [cs].

[23] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, N. Fiedel, PaLM: Scaling Language Modeling with Pathways, 2022. URL: http://arxiv.org/abs/2204.02311, arXiv:2204.02311 [cs].

[24] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, G. Lample, LLaMA: Open and Efficient Foundation Language Models, 2023. URL: http://arxiv.org/abs/2302.13971, arXiv:2302.13971 [cs].

[25] D. Ashok, Z. C. Lipton, PromptNER: Prompting For Named Entity Recognition, 2023. URL: http://arxiv.org/abs/2305.15444, arXiv:2305.15444 [cs].

[26] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, D. Zhou, Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, in: Advances in Neural Information Processing Systems, arXiv, 2022. URL: http://arxiv.org/abs/2201.11903, arXiv:2201.11903 [cs].

[27] S. Wang, X. Sun, X. Li, R. Ouyang, F. Wu, T. Zhang, J. Li, G. Wang, GPT-NER: Named Entity Recognition via Large Language Models, 2023. URL: http://arxiv.org/abs/2304.10428, arXiv:2304.10428 [cs].

[28] Q. Dong, L. Li, D. Dai, C. Zheng, J. Ma, R. Li, H. Xia,

J. Xu, Z. Wu, B. Chang, X. Sun, L. Li, Z. Sui, A Survey on In-context Learning, 2024. URL: http://arxiv.org/abs/2301.00234, arXiv:2301.00234 [cs].

[29] S. Pradhan, A. Moschitti, N. Xue, H. T. Ng, A. Björkelund, O. Uryupina, Y. Zhang, Z. Zhong, Towards Robust Linguistic Analysis using OntoNotes, in: J. Hockenmaier, S. Riedel (Eds.), Proceedings of the Seventeenth Conference on Computational Natural Language Learning, Association for Computational Linguistics, Sofia, Bulgaria, 2013, pp. 143–152. URL: https://aclanthology.org/W13-3516.

[30] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, in: H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, volume 33, Curran Associates, Inc., 2020, pp. 9459–9474. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf.

[31] S. Wang, Y. Meng, R. Ouyang, J. Li, T. Zhang, L. Lyu, G. Wang, GNN-SL: Sequence Labeling Based on Nearest Examples via GNN, in: A. Rogers, J. Boyd-Graber, N. Okazaki (Eds.), Findings of the Association for Computational Linguistics: ACL 2023, Association for Computational Linguistics, Toronto, Canada, 2023, pp. 12679–12692. URL: https://aclanthology.org/2023.findings-acl.803. doi:10.18653/v1/2023.findings-acl.803.

[32] D. Batra, N. Diwan, U. Upadhyay, J. S. Kalra, T. Sharma, A. K. Sharma, D. Khanna, J. S. Marwah, S. Kalathil, N. Singh, R. Tuwani, G. Bagler, RecipeDB: a resource for exploring recipes, Database 2020 (2020) baaa077. URL: https://doi.org/10.1093/database/baaa077. doi:10.1093/database/baaa077.

[33] D. M. Dooley, E. J. Griffiths, G. S. Gosal, P. L. Buttigieg, R. Hoehndorf, M. C. Lange, L. M. Schriml, F. S. L. Brinkman, W. W. L. Hsiao, FoodOn: a harmonized food ontology to increase global food traceability, quality control and data integration, npj Science of Food 2 (2018) 23. URL: https://www.nature.com/articles/s41538-018-0032-6. doi:10.1038/s41538-018-0032-6.

[34] S. Haussmann, O. Seneviratne, Y. Chen, Y. Ne'eman, J. Codella, C.-H. Chen, D. L. McGuinness, M. J. Zaki, FoodKG: A Semantics-Driven Knowledge Graph for Food Recommendation, in: C. Ghidini, O. Hartig, M. Maleshkova, V. Svátek, I. Cruz, A. Hogan, J. Song, M. Lefrançois, F. Gandon (Eds.), The Semantic Web – ISWC 2019, Springer International Publishing, Cham, 2019, pp. 146–162. doi:10.1007/978-3-030-30796-7_10.

[35] J. Marin, A. Biswas, F. Ofli, N. Hynes, A. Salvador, Y. Aytar, I. Weber, A. Torralba, Recipe1M+: A Dataset for Learning Cross-Modal Embeddings for Cooking Recipes and Food Images, 2019. URL: http://arxiv.org/abs/1810.06553. doi:10.48550/arXiv.1810.06553, arXiv:1810.06553 [cs].

[36] M. Bień, M. Gilski, M. Maciejewska, W. Taisner, D. Wisniewski, A. Lawrynowicz, RecipeNLG: A Cooking Recipes Dataset for Semi-Structured Text Generation, in: B. Davis, Y. Graham, J. Kelleher, Y. Sripada (Eds.), Proceedings of the 13th International Conference on Natural Language Generation, Association for Computational Linguistics, Dublin, Ireland, 2020, pp. 22–28. URL: https://aclanthology.org/2020.inlg-1.4. doi:10.18653/v1/2020.inlg-1.4.

[37] K. S. Komariah, A. T. Purnomo, A. Satriawan, M. O. Hasanuddin, C. Setianingsih, B.-K. Sin, SMPT: A Semi-Supervised Multi-Model Prediction Technique for Food Ingredient Named Entity Recognition (FINER) Dataset Construction, Informatics 10 (2023) 10. URL: https://www.mdpi.com/2227-9709/10/1/10. doi:10.3390/informatics10010010, number: 1 Publisher: Multidisciplinary Digital Publishing Institute.

[38] A. Agarwal, J. Kapuriya, S. Agrawal, A. V. Konam, M. Goel, R. Gupta, S. Rastogi, N. Niharika, G. Bagler, Deep Learning Based Named Entity Recognition Models for Recipes, in: N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, N. Xue (Eds.), Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), ELRA and ICCL, Torino, Italia, 2024, pp. 4542–4554. URL: https://aclanthology.org/2024.lrec-main.406.

[39] C. Radu, C.-E. Staicu, L.-M. Mitrică, M. Dînşoreanu, R. Potolea, C. Lemnaru, Extracting Settings from Multilingual Recipes with Various Sequence Tagging Models: an Experimental Study, in: 2022 IEEE 18th International Conference on Intelligent Computer Communication and Processing (ICCP), 2022, pp. 65–72. URL: https://ieeexplore.ieee.org/document/10053968/?arnumber=10053968. doi:10.1109/ICCP56966.2022.10053968, iSSN: 2766-8495.

[40] A. Wróblewska, A. Kaliska, M. Pawłowski, D. Wiśniewski, W. Sosnowski, A. Ławrynowicz, TASTEset – Recipe Dataset and Food Entities Recognition Benchmark, 2022. URL: http://arxiv.org/abs/2204.07775.

[41] T. Kudo, J. Richardson, SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing, in: E. Blanco, W. Lu (Eds.), Proceedings of the 2018 Conference on Empirical Methods in Natural Language Pro-

cessing: System Demonstrations, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 66–71. URL: https://aclanthology.org/D18-2012. doi:10.18653/v1/D18-2012.

[42] P. Blunsom, Hidden markov models, Lecture notes, August 15 (2004) 48.

[43] P. He, J. Gao, W. Chen, Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, 2021. arXiv:2111.09543.

[44] P. Rajpurkar, R. Jia, P. Liang, Know what you don't know: Unanswerable questions for SQuAD, in: I. Gurevych, Y. Miyao (Eds.), Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 784–789. URL: https://aclanthology.org/P18-2124. doi:10.18653/v1/P18-2124.

[45] S. Schweter, J. Baiter, Dbmdz BERT Models, https://github.com/dbmdz/berts, 2019. Accessed: 2024-04-22.

[46] L. A. Ramshaw, M. P. Marcus, Text Chunking using Transformation-Based Learning, 1995. URL: http://arxiv.org/abs/cmp-lg/9505040. doi:10.48550/arXiv.cmp-lg/9505040, arXiv:cmp-lg/9505040.

[47] J. Cañete, G. Chaperon, R. Fuentes, J.-H. Ho, H. Kang, J. Pérez, Spanish pre-trained bert model and evaluation data, in: PML4DC at ICLR 2020, 2020.

[48] H. Schwenk, V. Chaudhary, S. Sun, H. Gong, F. Guzmán, WikiMatrix: Mining 135M parallel sentences in 1620 language pairs from Wikipedia, in: P. Merlo, J. Tiedemann, R. Tsarfaty (Eds.), Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, Association for Computational Linguistics, Online, 2021, pp. 1351–1361. URL: https://aclanthology.org/2021.eacl-main.115. doi:10.18653/v1/2021.eacl-main.115.

A. Incorporating Spanish

In order to test more thoroughly the soundness of our approach, we carry out an equivalent study with Spanish.

A.1. Data

We annotated an English–Spanish dataset of recipes obtained from My Colombian Recipes,[11] which we refer to as MCR. MCR is translated from English to Spanish, which is evident from the fact that on the website all Spanish recipes have an English counterpart, but not vice versa. We believe approximately 5-10% of the dataset's instances to be possible MT. A good indication of this is the fact that the English "to taste" is sometimes translated as "para probar", likely an MT mistake, while other times the correct "al gusto" is used. Although using machine-translated data is not ideal, this was our best choice for a Spanish-language parallel recipe corpus, due to the lack of availability of similar online resources. The use of MT data has implications with respect to the evaluation of the models, as their performance would likely be lower in a real-world scenario involving recipes written directly in Spanish. Nonetheless, given the limited amount of data we hypothesize as being machine-translated, we believe the impact would not be large enough to discredit our results, which focus on the improvement over the cross-lingual EN–ES baseline, rather than the absolute performance of the best model.

MCR contains 276 recipes, 104 of which are bilingual and annotated with alignments. Due to this imbalance between the number of English and Spanish recipes, the number of entities is around 3x for the former, as shown in Table 5. In total, MCR contains annotations for 15,257 entities and 3,565 alignments. Along with the ingredient lists, MCR also contains cooking instructions for all its recipes, along with nutritional facts for 139 of them.

A.2. BERT Model

As a monolingual Spanish BERT model baseline to compare against mBERT, we use `bert-base-spanish-wwm-cased` ("BERT$_{\text{es}}$") [47].

A.3. Results

Entity Alignment Table 6 reports the results for the alignment task, complete with the settings including Spanish-language data.

Fine-tuning on the same language as the test set yields better results than cross-lingual scenarios. Furthermore, the best performance on MCR is obtained when fine-tuning mDeBERTa$_{\text{X}}$ on both Italian and Spanish.

This is not the case for mBERT$_{\text{X}}$ and mDeBERTa, whose performance is hindered by the addition of Italian training data. MCR is much narrower in terms of culinary variety, focusing solely on Colombian recipes. On the other hand, GZ contains not just traditional Italian recipes, but an international range of dishes. This is probably the reason why bilingual training is helpful on GZ, but is not beneficial with relation to MCR: adding data from a separate locale helps the models when approaching the more varied GZ, helping them generalize more effectively over its data. Conversely, they are thrown off

[11] https://www.mycolombianrecipes.com

	TS / EMT	**SMT**				**GZ**		**MCR**	
		mBERT$_X$		mDeBERTa$_X$					
Class	en / it / es	it	es	it	es	en	it	en	es
food	4,020	3,999	4,012	4,017	4,018	5,958	6,473	3,600	1,143
quantity	3,780	3,764	3,778	3,777	3,780	10,186	6,564	2,945	962
unit	3,172	3,151	3,169	3,159	3,171	8,148	4,450	2,325	760
process	1,091	1,066	1,089	1,090	1,091	217	265	1,236	379
physical q.	793	785	791	791	793	1,245	1,547	897	285
color	231	226	231	231	231	482	479	309	97
taste	126	121	123	125	123	98	72	8	2
purpose	94	94	94	94	94	69	126	89	34
part	55	53	55	55	55	220	263	142	44
total	**13,362**	**13,259**	**13,342**	**13,339**	**13,356**	**26,631**	**20,272**	**11,551**	**3,706**

Table 5
Dataset class distributions. EMT and SMT refer to the entity- and sentence-wise machine-translated TASTEset. GZ and MCR refer to our testing datasets.

Data	***P***	**mBERT**		**mBERT$_X$**		**mDeBERTa**		**mDeBERTa$_X$**	
		GZ	MCR	GZ	MCR	GZ	MCR	GZ	MCR
EMT									
it	0	35.93±0.79		38.87±0.48		42.17±1.19		46.98±3.77	
	0.1	43.13±2.51		44.49±1.21		57.00±0.94		58.45±1.37	
	0.2	42.54±1.37		44.02±3.32		55.03±2.40		57.02±2.43	
es	0		49.03±0.59		50.38±1.10		51.93±0.65		53.20±0.83
	0.1		63.69±0.96		67.60±0.74		**70.43±2.48**		71.07±3.62
	0.2		66.07±1.30		**70.20±1.66**		69.25±1.94		72.62±1.93
it–es	0	33.82±5.30	46.59±0.98	41.54±1.87	47.72±1.56	46.98±3.77	40.33±1.97	45.17±2.58	52.70±1.09
	0.1	43.36±2.72	64.57±2.35	46.14±3.85	67.16±2.19	**58.45±1.37**	53.68±2.45	57.64±0.83	**72.95±1.75**
	0.2	**44.37±1.57**	**67.62±0.33**	**47.14±1.43**	69.10±1.10	57.02±2.43	54.87±1.37	**58.84±2.68**	72.71±1.83
XL-WA									
it	–			21.04				31.71	
es	–				54.14			58.56	
it–es	–			23.60	53.89			33.56 70.47	

Table 6
Alignment task results (Exact metric). All results are averaged out of 5 random runs, besides the XL-WA baselines. Best in bold.

by the addition of out-of-domain data when tested on MCR's narrower domain.

Comparing the EMT fine-tuning results with the baselines at the bottom of Table 6, we can see that further fine-tuning on EMT does provide a boost, compared to training only on XL-WA. Nonetheless, the difference in performance is much greater when testing on GZ, compared to MCR. When looking at mBERT$_X$, fine-tuned on both Italian and Spanish, the model improves by more than 23 Exact points on GZ, while the gap in performance is just under 16 points on MCR. This effect is even more dramatic for mDeBERTa$_X$, with a difference of more than 25 points on GZ, but only 2.48 points on MCR.

Compounded with the fact that, in general, the metrics are much higher when testing on MCR compared to GZ, this points to MCR being a much less challenging test set, compared to GZ. As previously mentioned, part of the dataset is likely machine translated, and since an MT engine is more likely to follow rigidly defined patterns compared to a human translator, this might play a role into the alignment task being easier on these data.

Table 7 reports the performance of the best overall models on each of the individual classes, on both GZ and MCR. Giza++ essentially matches mDeBERTa's performance on MCR, which once again points to entities in MCR being easier to identify compared to GZ. However, the similar performance is largely due to mDeBERTa performing poorly on the `unit` and `part` classes, due to the reasons outlined in Section 4.

Entity Recognition Table 8 reports the results for the NER task for all language settings. For each language, we use the aligner models which obtained the highest results on the entity alignment task. Note that, since the aligner performance does not significantly improve with increased shuffling (see Section 5), we only train aligner models up to $P = 0.2$ for the Spanish setting due to computational constraints.

In the Spanish monolingual setting, both BERT$_{es}$ and mBERT obtain F$_1$ scores between 0.92 and 0.95 when fine-tuned on SMT, with the models fine-tuned on EMT trailing behind by 11 to 12 points. As all the models perform similarly and the standard deviation is also close to zero, it once again appears that the entities contained in the MCR dataset are not too challenging for both the mono- and multilingual models to identify.

Model	Test set	Qty.	Unit	Food	Process	Color	Phys. q.	Taste	Purpose	Part	Macro avg.
Fast Align	GZ	18.41	**30.94**	61.95	15.27	33.70	39.00	0.00	25.64	52.48	30.82
Fast Align	MCR	54.27	71.82	62.73	42.77	66.67	45.68	0.00	58.82	40.00	49.20
Giza++	GZ	35.21	15.24	77.01	51.91	84.81	71.76	27.03	61.54	**63.37**	54.21
Giza++	MCR	90.29	89.31	76.93	76.30	79.76	75.72	50.00	82.35	68.57	76.58
mBERTx$_{en-it-es}$	GZ	30.09	24.81	81.66	62.60	67.04	61.41	35.14	**94.87**	13.86	52.38
mBERTx$_{en-it-es}$	MCR	95.30	3.93	89.32	81.72	87.50	77.02	100.00	100.00	9.52	71.59
mDeBERTax$_{en-it-es}$	GZ	**54.95**	29.75	**83.49**	**83.21**	**85.93**	**87.66**	**75.68**	89.74	14.85	**67.25**
mDeBERTa	MCR	97.05	11.25	90.48	93.91	94.32	93.95	100.00	97.06	14.29	76.92

Table 7
Results of the alignment task by class for the best models, using the Exact metric. Best on GZ in bold, best on MCR underlined.

In the bilingual fine-tuning scenario, the training data is a concatenation of the SMT datasets produced by the models obtaining the highest performance on the two test sets. Since this is a bilingual fine-tuning scenario, we only use mBERT, as the monolingual models would not be able to be fine-tuned appropriately on this multilingual data. In this setup, the usefulness of the BERT-based aligners becomes more evident. Indeed, while performance on MCR is largely similar to the other setups, with all models outperforming the baseline by a large amount, the same cannot be said for mBERT's performance on GZ. Fine-tuning mBERT on the combination of the Italian and Spanish data aligned by Fast Align and Giza++ makes the NER model considerably worse at identifying entities in GZ, with a performance decrease of 20 F_1 points with the data created by Fast Align and of 21 F_1 points with that created by Giza++. The opposite is true when fine-tuning the mBERT NER model on the SMT data created by mDeBERTa, with the model achieving an F_1 of 0.94, beating the baseline by 5 points. Compared to the model fine-tuned on data created by Giza++, this represents a 26 F_1 point increase in performance.

As regards the baseline model fine-tuned on TASTEset's English data and tested on MCR's Spanish entities, we can see that, unexpectedly, the model obtains a 0.88 F_1 score, outperforming the mBERT (0.83 F_1) and BERT$_{es}$ (0.84 F_1) models fine-tuned on the monolingual Spanish EMT data. Despite this, fine-tuning on SMT data produced through our alignment approach allows the NER models to beat this 0.88 F_1 baseline, reaching scores as high as 0.95 F_1, as previously mentioned.

In all three scenarios, mBERT achieves performances comparable to those of the monolingual models. This shows that, when inferring on multilingual corpora to extract entities, a single multilingual model can be used, saving time and computational resources both during training and inference.

Train	Test	Aligner	NER	F_1
it	it	–	mBERT	0.89±0.01
		mBERT		0.91±0.02
		mDeBERTa		**0.94±0.01**
		Fast Align		0.84±0.01
		Giza++		0.87±0.03
		–	BERT$_{it}$	0.86±0.01
		mBERT		0.9±0.04
		mDeBERTa		**0.94±0.0**
		Fast Align		0.85±0.04
		Giza++		0.91±0.03
es	es	–	mBERT	0.83±0.01
		mBERT		**0.95±0.0**
		mDeBERTa		0.92±0.01
		Fast Align		0.94±0.0
		Giza++		**0.95±0.0**
		–	BERT$_{es}$	0.84±0.0
		mBERT		**0.95±0.0**
		mDeBERTa		0.93±0.01
		Fast Align		**0.95±0.0**
		Giza++		**0.95±0.0**
it–es	it	–	mBERT	0.89±0.01
		Fast Align		0.69±0.01
		Giza++		0.68±0.03
		mDeBERTa		**0.94±0.01**
	es	–	mBERT	0.83±0.0
		Fast Align		**0.95±0.0**
		Giza++		**0.95±0.0**
		mDeBERTa		0.94±0.01
en	it	–	mBERT	0.79±0.05
	es			0.88±0.01
	en (GZ)			0.9±0.01
	en (MCR)			**0.93±0.0**
	en (GZ)		BERT$_{en}$	0.91±0.01
	en (MCR)			**0.93±0.0**

Table 8
Entity recognition task F_1 scores (5 random runs macro avg).

B. XL-WA

As additional data for intermediate word-alignment training, we use XL-WA [16], a multilingual word-alignment dataset [16] built from WikiMatrix [48], [12] featuring 14 EN–XX language combinations. Its training set is composed of silver labels generated by a statistical model, while the development and test sets are manually annotated. Since XL-WA has a balanced domain distribution and can be considered representative of general language, it can be a good resource on which to train a baseline word-alignment model. Table 9 reports statistics for the EN–IT and EN–ES partitions used in this study.

[12] https://ai.meta.com/blog/wikimatrix/

	Sentences		**Alignments**	
Language	Train	Dev	Train	Dev
en–it	1,002	103	20,525	1,961
en–es	1,002	105	16,720	1,980

Table 9
Statistics for XL-WA's EN–IT and EN–ES subsets.

C. Computational Resources

All models are trained on a single NVIDIA RTX 5000 Ada Generation, with 32 GB of VRAM. The total training time is around 7-15 minutes for each alignment model, depending on the training data combination, plus 30-60 minutes for training each on XL-WA. Training each NER model takes around 6-7 minutes. All the training, including multiple models for standard deviation calculation, was carried out in under 48 hours.