Deep Learning Based Named Entity Recognition Models for Recipes

Mansi Goel[1,2,†], **Ayush Agarwal**[3,†], **Shubham Agrawal**[3,†], **Janak Kapuriya**[3,†], **Akhil Vamshi Konam**[3,†], **Rishabh Gupta**[3], **Shrey Rastogi**[3], **Niharika**[3] **and Ganesh Bagler**[1,2]

[1]Infosys Centre of Artificial Intelligence, IIIT-Delhi
[2]Department of Computational Biology, IIIT-Delhi
[3]Department of Computer Science, IIIT-Delhi
{mansig, ayush22095, shubham22124, kapuriya22032,
konam20513, rishabh21070, shrey21145, niharika21132, bagler}@iiitd.ac.in

Abstract

Food touches our lives through various endeavors, including flavor, nourishment, health, and sustainability. Recipes are cultural capsules transmitted across generations via unstructured text. Automated protocols for recognizing named entities, the building blocks of recipe text, are of immense value for various applications ranging from information extraction to novel recipe generation. Named entity recognition is a technique for extracting information from unstructured or semi-structured data with known labels. Starting with manually-annotated data of 6,611 ingredient phrases, we created an augmented dataset of 26,445 phrases cumulatively. Simultaneously, we systematically cleaned and analyzed ingredient phrases from RecipeDB, the gold-standard recipe data repository, and annotated them using the Stanford NER. Based on the analysis, we sampled a subset of 88,526 phrases using a clustering-based approach while preserving the diversity to create the machine-annotated dataset. A thorough investigation of NER approaches on these three datasets involving statistical, fine-tuning of deep learning-based language models and few-shot prompting on large language models (LLMs) provides deep insights. We conclude that few-shot prompting on LLMs has abysmal performance, whereas the fine-tuned spaCy-transformer emerges as the best model with macro-F1 scores of 95.9%, 96.04%, and 95.71% for the manually-annotated, augmented, and machine-annotated datasets, respectively.

Keywords: Named Entity Recognition, Deep Learning, Large Language Models, Language Modelling, Corpus, Language Representation Models, Information Extraction

1. Introduction

Food plays a central role in our lives. Beyond its primary purpose of nourishment and taste, it encompasses a broad spectrum of endeavors touching on health and sustainability. In the modern culinary landscape, where food is not just sustenance but reflects our diverse tastes and interests, information extraction in food texts has become increasingly crucial. As we explore culinary experiences and adapt to dietary preferences, extracting valuable information from food-related texts empowers us to make informed choices. Information extraction (Wei et al., 2023) enables efficient utilization of food-related data. This includes identifying ingredients and nutritional details in recipes (Kalra et al., 2020), ensuring dietary safety by detecting allergens (Pellegrini et al., 2021), optimizing restaurant operations through menu analysis (Syed and Chung, 2021), enhancing food safety by tracking recalls, cost and sustainability. These technological enhancements provide deeper perspectives on what we eat and facilitate personalized meal planning, culinary research, and innovation in the food industry.

Recipes are unstructured text, and named entities are their building blocks. Named entity recognition (NER) is a technique for extracting information from unstructured or semi-structured data with known labels (Chieu and Ng, 2002). It requires the many-to-one mapping of various named entities in text to their domain-specific categories. NER can extract information from various domains, including reviews, news articles, scientific literature, and food texts. NER not only acts as a standalone tool for information extraction but also plays an essential role in a variety of natural language processing (NLP) applications such as text understanding (Zhang et al., 2020; Cheng and Erk, 2020), information retrieval (Guo et al., 2009; Petkova and Bruce Croft, 2007), automatic text summarization (C. Aone and Larsen, 1999), question answering (Mollá et al., 2006), machine translation (Babych and Hartley, 2003), and knowledge base construction (Etzioni et al., 2005), etc. Recent studies have implemented various deep-learning models, such as BERT (Liu and Cui, 2023; Fang et al., 2023; Suleman et al., 2022), DistilBERT (Sanh et al., 2019; Hossain et al., 2022; Silalahi et al., 2022), DistilRoBERTa (Davidson et al., 2021; Qu et al., 2023; Rodrigues et al., 2022), spaCy (Kumar, 2023), and flair (Mathis, 2022; Pathak et al., 2022; Kumar et al., 2023).

Traditional NER models, such as Hidden Markov models (Rabiner and Juang, 1986) and conditional

[†]These authors contributed equally to this work.

LREC-COLING 2024, pages 4542–4554
20-25 May, 2024.

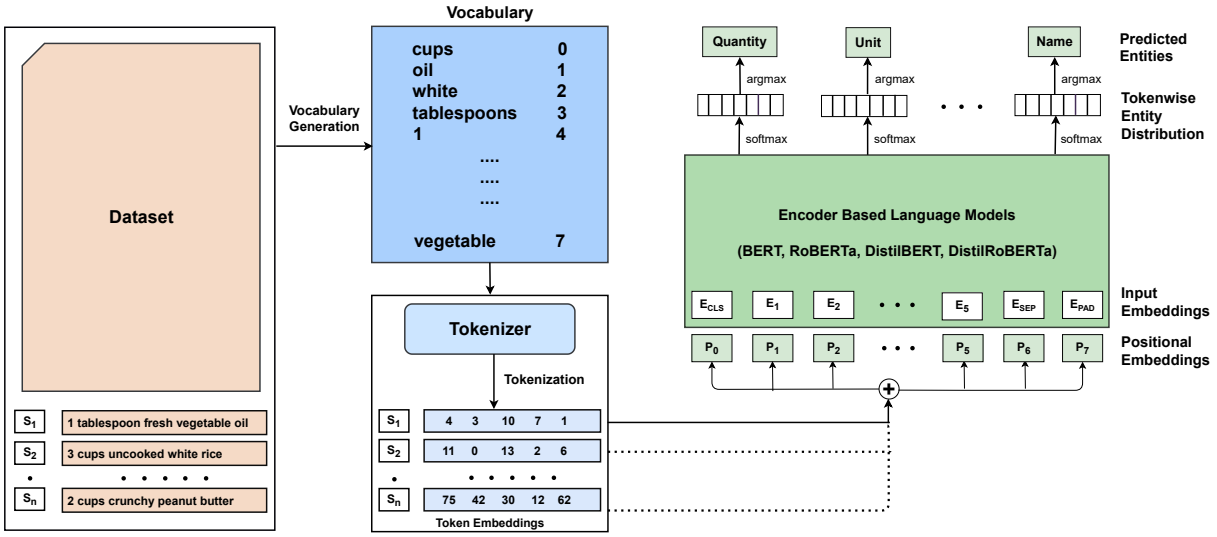

Figure 1: The pipeline implemented for fine-tuning supervised deep learning-based named entity recognition comprises three phases. To begin with, we built vocabularies for each of the three datasets. Further, we utilized these vocabularies to convert every word of an input sentence into corresponding token indexes that were subsequently converted to token embeddings via tokenization. Finally, Encoder-Only language models were employed to predict entity tags for the input token embeddings. The spaCy-transformer emerged as the best model with F1 scores of 95.9%, 96.04%, and 95.71% for Manually_Annotated, Augmented, and Machine_Annotated datasets, respectively.

random fields (Lafferty et al., 2001), rely heavily on rule-based features (Luo et al., 2015; Passos et al., 2014). DrNER (Eftimov et al., 2017) in rule-based NER that can extract food entities from evidence-based dietary recommendations. This work was extended to develop another rule-based NER FoodIE (Popovski et al., 2019), where the rules incorporate computational linguistics information. FoodIE achieved promising results on independent benchmark datasets and has been used to create the FoodBase corpus, the first NER corpus in the food domain. The limitation of the FoodIE method is its dependency on external resources, which have become inaccessible after its publication, rendering the method unusable. With a similar spirit, a data-driven method to find named entities, BuTTER (Cenikj et al., 2020), was trained on the FoodBase corpus based on Bidirectional Long Short-Term Memory and conditional random field methods. Radu et al. (2022) implemented NER on cooking instructions from multilingual recipes (French, German, and English). They implemented a Conditional Random Field layer on top of Bidirectional Long-Short Term Memory models, achieving F1 scores over 96% in mono and multi-lingual contexts for all classes. Another research (Brahma et al., 2020) implemented a NER approach to identify the food quality descriptors from chats between customers and customer support staff. Previous research (Diwan et al., 2020) used the RecipeDB dataset (Batra et al., 2020) to identify the named entities in ingredient phrases and cooking instructions. They reported an F1 score of 0.95 (ingredient), 0.88 (processes), and 0.90 (utensils). SciFood-NER (Cenikj et al., 2022) is a BERT-based method for recognizing named entities in scientific texts and achieved an F1 score of 0.90. NER can accurately identify ingredient names, quantities, unit, state, size, dry/fresh, and temperature within recipes and food-related content.

Computational Gastronomy represents the study of food, flavors, nutrition, health, and sustainability from the computing perspectives (Goel and Bagler, 2022). This new data science niche dramatically changes the outlook on food and cooking, traditionally considered artistic endeavors. In this context, building NER models for recipe texts is an exciting proposition, given its applications spanning multiple domains, including disease prediction, cost estimation, flavor profiling, and comprehensive nutritional analysis of recipes. Herein, we present a computational pipeline by utilizing encoder-based language models to extract NERs from recipe text (Figure 1).

The salient contributions of research studies presented here are (a) the introduction of augmented and machine-annotated ingredient phrase datasets, (b) analysis of the distribution of RecipeDB ingredient phrases, and (c) a thorough investigation of NER approaches on recipe texts involving statistical, deep-learning-based fine-tuning of language models and few-shot prompting on LLMs.

2. Dataset

We have used the manually annotated data consisting of 6,611 ingredient phrases (Diwan et al., 2020) that were sourced from RecipeDB (Batra et al., 2020), where all named entities were manually labeled (Manually_Annotated_Dataset). An augmented dataset comprising 26,445 ingredient phrases was created by label-wise token replacement, synonym replacement, and shuffling with segments (Augmented_Dataset).

We created an extensive repository of 349,762 unique ingredient phrases from the RecipeDB dataset (Batra et al., 2020) involving semi-automated processing protocol and human curation (Machine_Annotated_Dataset). These ingredient phrases were divided into 2,067 clusters (Stratified Entity Frequency Sampling) based on seven named entity tags (name, quantity, unit, df (dry/fresh), state, size, and temp) and 25% of data (88,526 phrases) were sampled for training. We used 2,187 Manually_Annotated_Dataset for testing.

2.1. Data Preprocessing

Starting with the 1,150,000 ingredient phrases obtained from RecipeDB dataset (Batra et al., 2020), we implemented a preprocessing protocol of lemmatization and manual annotations. A team of culinary experts manually identified the most frequent error patterns present in the dataset (see B). These mistakes were collectively rectified using Python scripts.

2.2. Data Augmentation

Language models need a larger dataset for training. Hence, to extend the Manually_Annotated_Dataset, we implemented three augmentation techniques (Figure 2).

Labelwise Token Replacement (LwTR): LwTR replaces the token with a random token from the training set with the same label after taking a call on whether a token should be replaced based on the binomial distribution. This procedure ensures that the original label sequence is preserved.

Synonym Replacement (SR): In a procedure analogous to LwTR, the SR method replaces the token randomly with its synonyms from the Wordnet lexical database.

Shuffle within Segments (SiS): In SiS, the token sequence is first split into segments with the same label, so each segment has some probability of shuffling (as per binomial distribution). The token within the same segment is then shuffled while keeping the order of tokens unchanged.

2.3. Machine-Annotated Dataset

We had a training dataset with 6,611 and 2,187 labeled ingredient phrases for training and testing. Given ten ingredients per recipe on average in a recipe, this yields around 661 recipes for training and 218 recipes for testing. These data are of limited utility when training transformer-based language models on which our experiments are based and which are known to excel in NLP tasks such as named entity recognition.

Dataset Creation

Given the size of the ingredient phrase corpus (1,150,000 ingredient phrases), it was deemed impractical to annotate the entire RecipeDB. After removing duplicates (an ingredient phrase may be a part of several recipes), we were left with 349,762 unique phrases. We adopted a hybrid approach to address this challenge. First, we trained the Stanford NER on the labeled corpus (6,611 + 2,187 = 8,798 Ingredients) to annotate the unique ingredient phrases from RecipeDB. Then, we manually cleaned the machine-generated annotations to identify the error patterns and correct them programmatically. We implemented Stratified Entity Frequency Sampling, a clustering and sampling approach, to sample 25% (88,526 phrases) of the unique ingredient phrases.

Stratified Entity Frequency Sampling

The unique challenges posed by our dataset led to the development of a clustering and sampling technique that we term 'Stratified Entity Frequency Sampling (SEFS).' SEFS ensures a diverse and representative selection of annotated data from a vast corpus, maximizing the capture of varied ingredient phrase patterns.

SEFS operates on the premise that ingredient phrases vary based on the combination and frequency of entities they contain. Some phrases may contain the ingredient's name only, while others could be more descriptive, indicating quantity, unit, state, size, and temperature. Ensuring a wide-ranging representation of these combinations in our sample was imperative to train a robust model.

Clustering The first step in SEFS is to cluster the unique ingredient phrases based on their entity composition. An entity frequency vector is created for each phrase, where each vector component represents the count of a specific entity (name, quantity, unit, state, size, or temperature) in the phrase. These vectors serve as the basis for clustering, where each unique vector corresponds to a cluster. This ensures that ingredient phrases with the same entity composition and frequency are grouped.

Sampling Once clustered, we sample from these

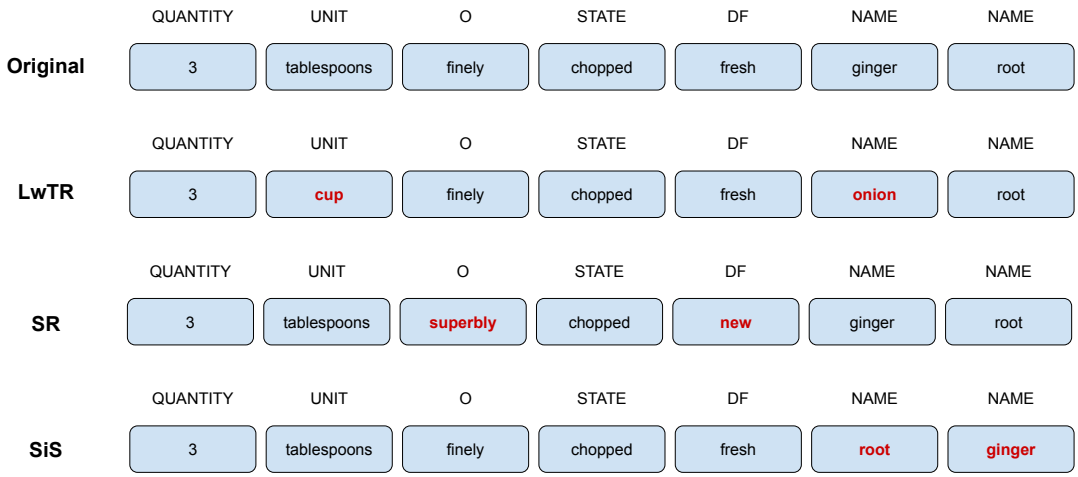

Figure 2: Illustration of Data Augmentation strategies to generate new samples. (a) LwTR: Labelwise Token Replacement: replace a token with a random token of the same label. (b) SR: Synonym Replacement: replace a token with its synonym from Wordnet. (c) SiS: Shuffle within segments: shuffle the tokens under their corresponding label within an ingredient phrase.

groups to create our dataset. A uniform sampling might not capture the richness and variability of the corpus. Therefore, we adopt a stratified sampling approach. In this method, we sample a fixed proportion (25%, in our case) from each cluster. This guarantees that the resultant dataset contains diverse ingredient phrase patterns.

SEFS ensures that our sample is not biased towards any particular type of ingredient phrase. It captures the breadth and diversity of the RecipeDB, making it particularly suited for training transformer-based models that thrive on varied data. Moreover, the stratified sampling ensures that even rarer patterns, which could be missed in a random sampling approach, are included in the dataset.

Figure 3 depicts the skewed distribution of ingredient phrases across clusters. Around 90% of the total ingredient phrases (1.15 million) can be represented by only 91 unique entity frequency vectors, and the remaining 10% of the phrases require 1,976 different frequency vectors for their representation. This shows that random sampling of ingredient phrases may lead to a bias towards the majority frequency vectors and justifies the SEFS sampling strategy.

3. Named Entity Recognition Models

3.1. Model Configurations

Building upon the previous work (Diwan et al., 2020), we re-implemented the Stanford-NER (Finkel et al., 2005). The Standford NER was trained using CRFClassifier with default parameters on an 8 GB CPU RAM system. We implemented diverse deep-learning NER models (BERT, DistilBERT, RoBERTa, and DistilRoBERTa) and NLP frameworks (spaCy, and flair) to find the named entities in the ingredients section. We fine-tuned our datasets on base-case variants of BERT, DistilBERT, RoBERTa, and DistilRoBERTa models with their pre-trained weights using an SGD optimizer with a learning rate 1e-2. All these models were run on an NVIDIA A100 80GB PCIe GPU card with a batch size of 44 and up to 12 epochs. We have used two different pipelines of spaCy 3.6.1 (en_core_web_lg - a classical rule-based NLP pipeline optimized for CPU, and en_core_web_trf - a RoBERTa-based transformer pipeline). Flair used a pre-trained xlm-roberta-large model to perform the NER.

3.2. Modelling Techniques

BERT (Devlin et al., 2019) captures the contextual nuances in language by considering the surrounding context of a word in a sentence. Apart from BERT, we employed its other three variants - DistilBERT (Sanh et al., 2019), RoBERTa (Liu and Cui, 2023) and DistilRoBERTa (Sanh et al., 2019). NLP frameworks such as spaCy (Matthew et al., 2020), flair (Akbik et al., 2019) have been implemented to find the named entities of ingredient phrases. A tool Stanford NER (Finkel et al., 2005) employs Conditional Random Fields to analyze and tag entities in a given text with their respective categories. One of its notable features is its ability to recognize and classify entities in multiple languages, making it valuable for multilingual applications.

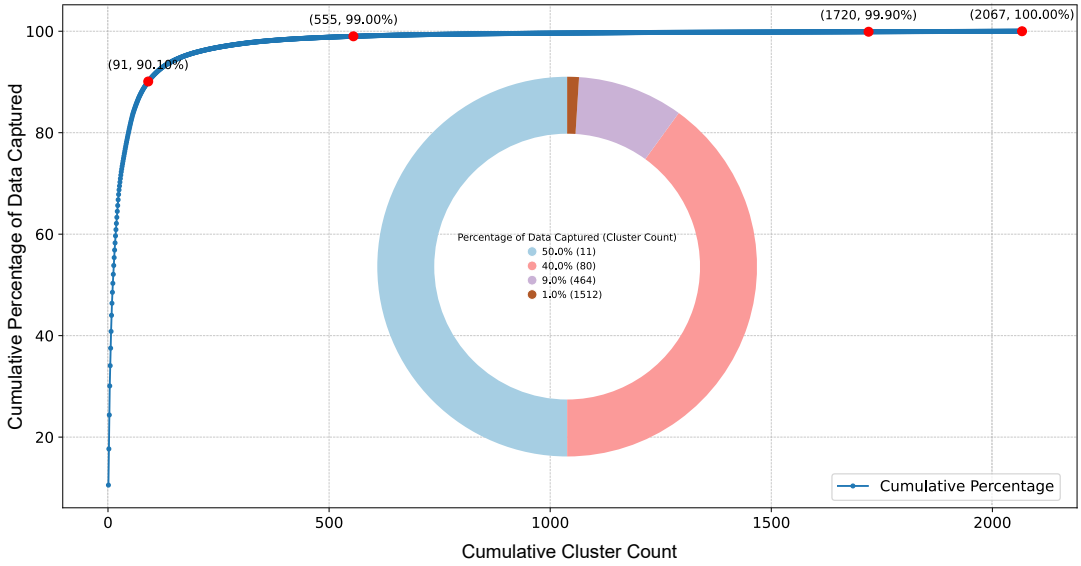

Figure 3: Analysis of the percentage of ingredient phrases captured by various clusters. The distribution is extremely skewed, with a few clusters hoarding most ingredient phrases. Half of the ingredient phrases, for example, are captured by merely the eleven largest clusters.

4. Model Evaluation

We employ macro-F1 score, precision, and recall to evaluate our models' predictive performance. These metrics address the inherent class imbalance in our datasets, where accuracy can be misleading. The F1 score provides a robust measure in such cases. Precision and recall are equally critical for our task, as we prioritize correctly identifying all valid ingredient tags (particularly names and quantities) without omissions. While the macro-F1 score is an average of tag-wise F1 scores, it's important to note that it doesn't directly follow the typical harmonic mean relationship with precision and recall. This is because macro-averaging calculates these metrics separately for each label and then averages them, giving equal weight to all labels – a crucial distinction for interpreting results in multi-label classification tasks.

5. Results

Pattern recognition aimed at NER across manual, augmented, and machine-annotated datasets is a difficult task due to degenerate tags corresponding to the same named entity. These ambiguous associations have origins in the linguistic subtleties referring to food's taste, value, and utility. For example, the word 'sour' in 'sour cream' signifies STATE, whereas in 'ice cream,' it collectively represents an ingredient; hence, both entities should belong to the NAME tag.

Herein, we present state-of-the-art models based on deep learning and statistical approaches for named entity recognition in recipe texts. This section is arranged as follows: Section 5.1 discusses the implementation of Stanford NER (Finkel et al., 2005), which uses statistical-based techniques for NER. In Section 5.2, we evaluate relevant deep-learning-based models fine-tuned on our datasets for performance. Section 5.3 describes the tag-wise analysis of named entities using the best performing model, and finally, Section 5.4 delves into the few-shot prompting experiments using state-of-the-art LLMS.

5.1. Stanford NER Implementation

We used the Stanford NER (Finkel et al., 2005), to reproduce the earlier work of Nirav et.al (Diwan et al., 2020) and have found consistent results (Table 1). We obtained the same results for seven out of nine experiments, and for the rest of the two, the deviation was <1%. These results signify the importance of CRF-based methods, which have been the go-to methods for recipe NER in most previous works (Diwan et al., 2020; Patil et al., 2020; Wei et al., 2016; Yang and Huang, 2018; Sato et al., 2017). By building on the learnings from these articles and rooted in extensive datasets introduced in this study, we implement deep-learning-based, state-of-the-art fine-tuned models.

5.2. Supervised Fine-tuning of Encoder-based Language Models

To enhance the performance of Named Entity Recognition on recipes, we began with a baseline model, Stanford NER (Finkel et al., 2005).

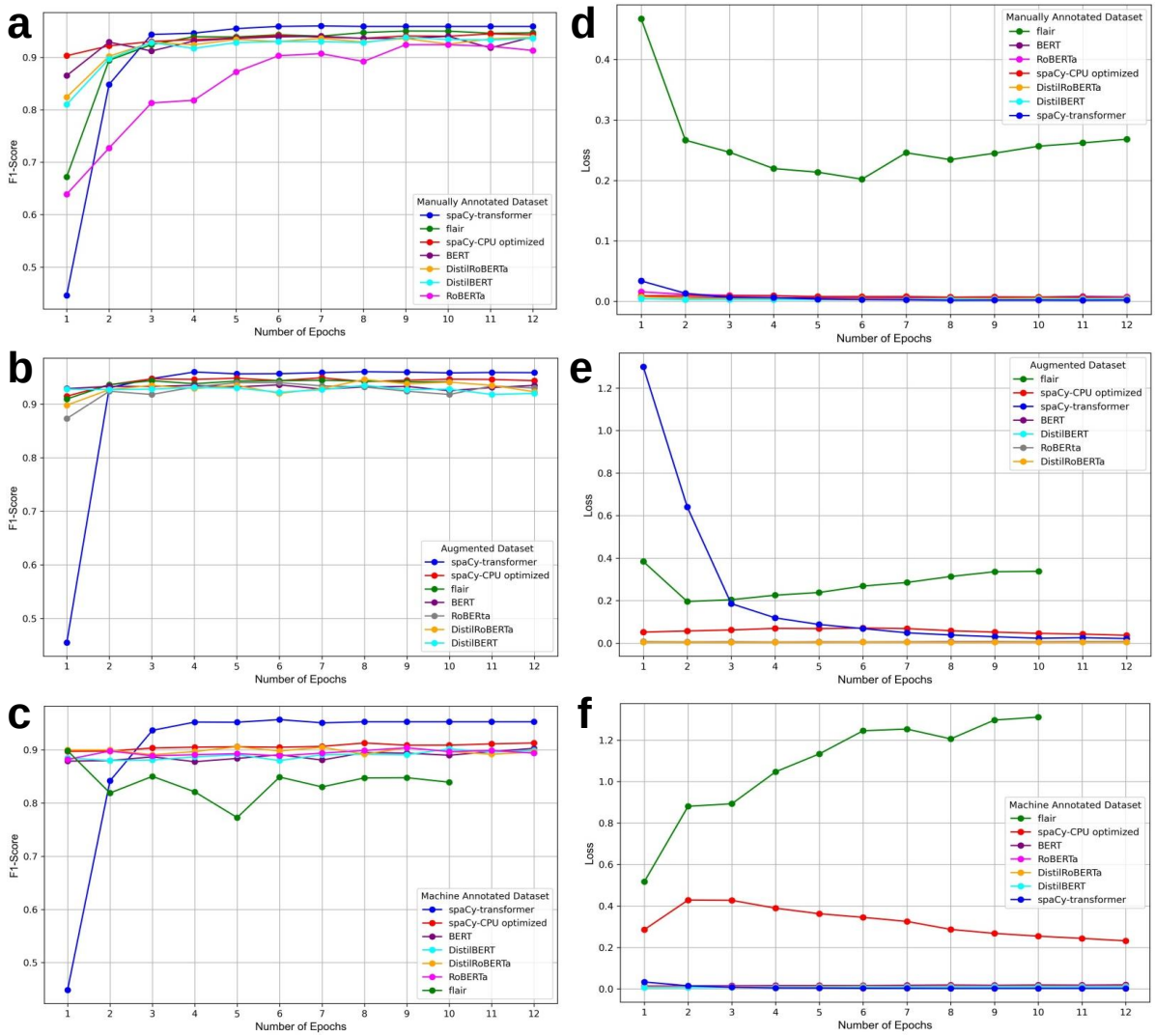

Figure 4: Model Comparison based on F1-scores and Loss. (a), (b) and (c) represent epoch-wise F1-score for Manually Annotated, Augmented, and Machine Annotated Datasets, respectively. Similarly, (d), (e) and (f) represent the epoch-wise Loss score for three datasets.

Test Set	Train Set					
	Diwan et al. (2020)			Present Study		
	AR	GK	Both	AR	GK	Both
AR	96.82	93.17	97.09	96.82	93.31	97.04
GK	86.72	95.19	94.98	86.71	95.16	95.02
Both	89.72	94.72	96.11	**89.16**	94.72	**95.52**

Table 1: Performance comparison of Diwan et al. (2020) and our implementation of Stanford NER. AllRecipes.com (AR) and geniuskitchen.com (GK) refer to the source of recipes from where the raw data was compiled to create the Manually_Annotated dataset.

We further implemented seven deep-learning models, including BERT variants (BERT, DistilBERT, RoBERTa, and DistilRoBERTa) and NLP toolkits (SpaCy with CPU optimization, SpaCy equipped with transformer, and flair). To ensure a comprehensive assessment, each model was fine-tuned across three distinct datasets before being consistently evaluated on the Manually Annotated test dataset of 2187 ingredient phrases (Diwan et al., 2020).

Figure 4 depicts epoch-wise F1 and validation loss scores for all three datasets across all models. Table 2 encapsulates the results from the best epoch for every dataset-model pair. Despite starting with

Modelling Technique	Manually_Annotated			Augmented			Machine_Annotated		
	F1 (%)	P (%)	R (%)	F1 (%)	P(%)	R (%)	F1 (%)	P(%)	R (%)
spaCy-transformer	**95.90**	**95.89**	**95.91**	**96.04**	**96.05**	**96.04**	**95.71**	**95.73**	**95.69**
spaCy-CPU optimized	94.46	94.52	94.41	94.91	94.92	94.90	91.30	91.36	91.24
Stanford NER	95.52	95.64	95.39	95.16	94.37	95.96	89.9	91.31	88.53
DistilBERT	93.80	95.20	93.60	93.50	93.50	94.60	90.20	92.20	89.70
BERT	94.00	94.70	94.10	93.60	93.70	94.10	90.30	91.50	90.20
DistilRoBERTa	93.80	94.80	93.90	94.60	94.10	95.90	90.60	91.60	90.60
RoBERTa	92.40	92.90	92.60	94.00	94.50	94.10	90.40	91.60	90.20
flair	95.01	96.11	96.05	94.45	95.87	96.14	89.85	88.71	89.22

Table 2: Performance Evaluation on Manually Annotated, Augmented and Machine Annotated Datasets

a lower F1 score, the spaCy-transformer exhibits a rapid learning curve, eventually surpassing the performances of its counterparts. Such discrepancies, especially during the initial epochs across various models, can be attributed to the inherent variability arising from the seed values of model weights and consistent hyperparameters employed. The Augmented dataset, as expected, shows slight performance gains, which is explained by the fact that DL models are data-hungry and given more examples, they would learn the entity-tag mapping better. However, the Machine_Augmented dataset with silver labels created using DL models previously trained on Manually_Annotated datasets appears to echo the inherent variability and noise, coupled with potential mislabeling. This explains a slight decrease in its performance compared to the manually annotated dataset.

A particularly captivating observation emerged from our analysis of the Distil-versions compared to their original BERT-based counterparts. Contrary to conventional assumptions, the Distil-variants held their ground and frequently outperformed the base models. This phenomenon merits a closer examination. Several plausible factors could be driving this unexpected outcome. Firstly, the base BERT variants might be predisposed to overfitting the peculiarities of the training set. Such a tendency would culminate in an escalated validation loss, suggestive of an overly tailored model struggling to generalize to new, unseen data.

Additionally, the presence of fine-grained, spurious correlations within the dataset could be more readily captured by these base models. While seemingly advantageous, this heightened sensitivity might be counterproductive by leading the model to internalize these inconsequential patterns as meaningful, skewing its predictions. Moreover, the potential presence of label noise within the datasets might cause Base BERT models to be overly adept at learning these noise-influenced labels. Consequently, while they might produce tags mirroring the original distribution, these tags might deviate from the expected results in the validation set, thereby being marked erroneous. Summarising, The Distil versions, being smaller with fewer parameters, are weaker in capturing the 'bad patterns'–spurious correlations and label noise, which surprisingly acts in favor of their performance metrics.

As we see from our results on the augmented dataset, some models perform better on the augmented datasets, such as spaCyNER and DistillRoBERTa. Because deep-learning-based language models are data-hungry, we enhanced the volume of our dataset by using data augmentation techniques. Consequently, the model performances get a boost as they get more examples to learn about the inherent nature of ingredient phrases.

Analysis of the results obtained in the previous section reveals that spaCy-transformer stands out as the best deep-learning-powered package for recognizing entity tags in recipe texts. It outperformed all other models and baselines on all three datasets. It has also shown stable, consistent learning for our models with the least variance compared to others, as shown in Figure 4.

5.3. Tag-wise Analysis of Named Entities

In our investigation of epoch-wise learning trends for various entity tags using our top-performing model, a notable correlation emerges between the frequency of a tag in the dataset and its learning trajectory within the model. Consistently, across all three datasets, the 'Quantity' tag exhibits the earliest and most robust learning, while the 'Temperature' tag lags, both in initiation and overall learning, as shown in Figure 5 by their F1 scores. This disparity underscores the model's limitations in grasping rare tags as effectively as with prevalent ones. A plausible interpretation of this observation is that while attempting semantic understanding, the models also rely on memorizing specific entity-tag pairings. Consequently, less frequent tags that offer fewer memorization opportunities tend to be underlearned.

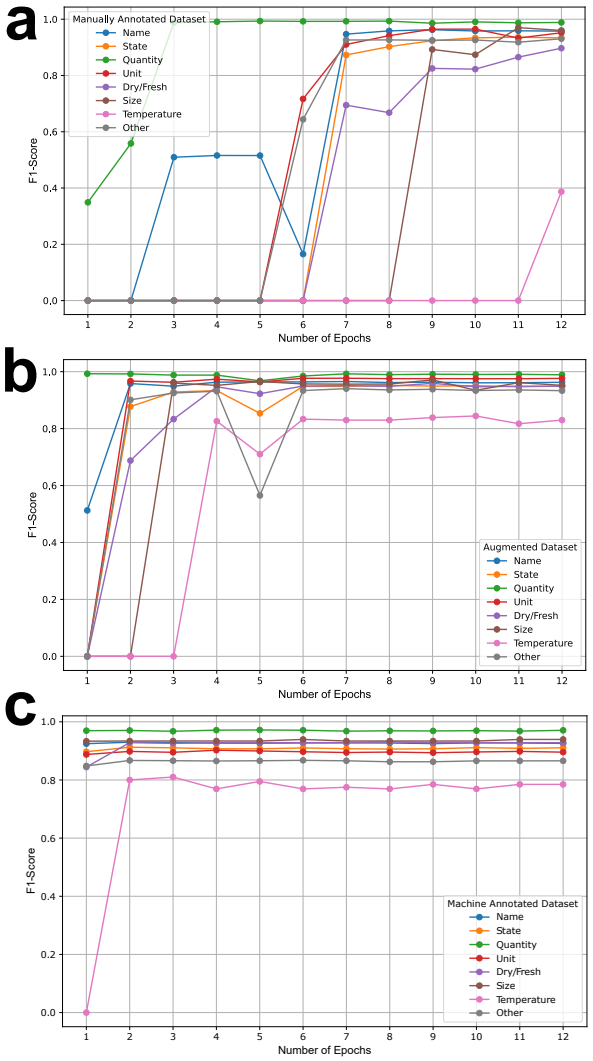

Figure 5: Tag-wise learnability of named entities and their final results using the best-performing model–the spaCy-transformer. Figures (a), (b) and (c) depict these results for the Manually_Annotated, Augmented, and Machine_Augmented datasets, respectively.

5.4. Analysis of Few-Shot Prompting on LLMs

Few-shot NER leverages the power of LLMs, such as Chat-GPT and GPT-4 (Wang et al., 2023; Ji, 2023), to tackle the challenging task of entity recognition with minimal annotated data. A prompt is given as input to the LLM that outlines the NER task and specifies the context and available examples (6). This prompt acts as a few-shot learning signal, enabling the model to understand the task and context. The pre-trained LLM predicts named entities in a given text. Few-shot NER is useful with limited labeled data, as it can quickly adapt to new entity types and domains. While fine-tuning specific data can further enhance performance, the strength of LLMs lies in their ability to perform remarkably well in a wide range of NLP tasks.

Table 3 indicates that pre-trained LLMs have limited exposure to food and culinary datasets during their initial pretraining. Consequently, their performance in in-context learning, especially in food-related named entity recognition, is suboptimal. This deficiency in domain-specific knowledge acquired during pretraining significantly affects their in-context learning capabilities and overall task performance. It underscores the need to fine-tune these models with domain-specific datasets to enhance their effectiveness in specialized tasks.

Model	Macro-F1 (%)	Micro-F1 (%)
LLaMA2-7b	5.88	44.29
LLaMA2-13b	17.06	54.20
Mistral-7b	32.78	47.51
Vicuna-7b	32.90	51.41

Table 3: Results of NER using Few-Shot Prompting on the state-of-the-art LLMs.

6. Discussion and Conclusions

Our study presented one of the most extensive labeled data resources of named entities from recipe ingredient phrases. Further, we present deep-learning and statistical models built to achieve state-of-the-art results. Nonetheless, our studies are limited in certain aspects of culinary context, nuances of data, and modeling paradigm.

Our present study focuses on only ingredient phrases while not accounting for the recipe instructions, which often carry semantic information about cooking that encodes cultural nuances. Further, static pre-trained models, such as BERT, RoBERTa, and XLM-RoBERTa, come with inherent biases and might not be fine-tuned to capture the nuances of the food lexicon. Complex culinary instructions may not be amenable to extracting meaningful information. For example, the phrase 'ground roasted peanuts' holds multiple layers of information, posing a severe challenge for NER. Names of ingredients unique to certain cuisines might be tokenized sub-optimally, leading to NER errors.

In the future, this research may be extended to include LLM fine-tuning, implementing NERs on cooking instruction, prompt engineering for LLMs for NER on recipes, soft prompt tuning, chain of thought, and implementation of multilingual NER.

7. Acknowledgements

GB thanks Infosys Center of Artificial Intelligence, Centre of Excellence in Healthcare, and IIIT-Delhi for the computational support. MG thanks IIIT-Delhi

Below is an instruction that describes a task, paired with an input that provides further context. Write a response that appropriately completes the request.

Instruction:
Your task is to do Named Entity Recognition of input sentence. You must assign entity tags to each word in given input sentence from
[QUANTITY, UNIT, NAME, TEMP, STATE, SIZE, DF, O].

Number of tokens in input and output sentences must be equal.

Where,
NAME is the name of the ingredient added into the recipe, like onion, garlic etc.
UNIT is the unitary amount of the ingredient added into each step of recipe, like cup, tablespoon, etc.
QUANTITY is a multiple of the UNIT tag which gives the total quantity of the ingredient used in every step of the recipe.
TEMP is the temperature based state of the ingredient, like frozen, hot etc.
STATE is the condition of the ingredient used, like chopped, ground etc.
SIZE is the qualitative amount of the ingredient in each step of the recipe.
DF is the Dry or Fresh condition of the ingredient.
O is Others which is used for entities which are none of these : [QUANTITY, UNIT, NAME, TEMP, STATE, SIZE, DF]

Some Examples:

Input: '2 tablespoons vegetable oil , divided'
Output: [QUANTITY, UNIT, NAME, NAME, O, STATE]

Input: '2 tablespoons dried marjoram'
Output: [QUANTITY, UNIT, DF, NAME]

Input: '1 -LRB- 12 ounce -RRB- box Barilla Gluten Free Penne'
Output: [QUANTITY, O, QUANTITY, UNIT, O, UNIT, NAME, NAME, NAME, NAME]

Input: '2 jalapeno peppers , seeded and minced'
Output: [QUANTITY, NAME, NAME, O, STATE, O, STATE]

Input:
{input_sentence}
Output:

Figure 6: The hand-crafted prompt given to LLMs during Few-Shot Prompting.

for the research fellowship. The authors thank Technology Innovation Hub (TiH) Anubhuti for the research grant.

8. Code and Data Availability

All relevant code files and datasets are available on GitHub.

9. Bibliographical References

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP. In *Association for Computational Linguistics*, pages 54–59, Minneapolis, Minnesota. Association for Computational Linguistics.

Bogdan Babych and Anthony Hartley. 2003. Improving machine translation quality with automatic named entity recognition. In *7th International EAMT workshop on MT and other language technology tools, Improving MT through other language technology tools, Resource and tools for building MT at EACL*, pages 1–8.

Devansh Batra, Nirav Diwan, Utkarsh Upadhyay, Jushaan Singh Kalra, Tript Sharma, Aman Kumar Sharma, Dheeraj Khanna, Jaspreet Singh Marwah, Srilakshmi Kalathil, Navjot Singh, Rudraksh Tuwani, and Ganesh Bagler. 2020. RecipeDB: A resource for exploring recipes. *Database*, page 77.

Aditya Kiran Brahma, Prathyush Potluri, Meghana Kanapaneni, Sumanth Prabhu, and Sundeep Teki. 2020. Identification of Food Quality Descriptors in Customer Chat Conversations using Named Entity Recognition. In *ACM International Conference Proceeding Series*, CODS-COMAD '21, pages 257–261, New York, NY, USA. Association for Computing Machinery.

J. Gorlinsky C. Aone, M. E. Okurowski and B. Larsen. 1999. A trainable summarizer with knowledge acquired from robust nlp techniques. *Advances in Automatic Text Summarization*, MIT Press:71–80+.

Gjorgjina Cenikj, Gasper Petelin, Barbara Korousic Seljak, and Tome Eftimov. 2022. SciFoodNER:

Food Named Entity Recognition for Scientific Text. In *Proceedings - IEEE International Conference on Big Data*, pages 4065–4073. IEEE.

Gjorgjina Cenikj, Gorjan Popovski, Riste Stojanov, Barbara Koroušić Seljak, and Tome Eftimov. 2020. Butter: Bidirectional lstm for food named-entity recognition. In *IEEE International Conference on Big Data (Big Data)*, pages 3550–3556. IEEE.

Pengxiang Cheng and Katrin Erk. 2020. Attending to Entities for Better Text Understanding. In *34th AAAI Conference on Artificial Intelligence*, volume 34, pages 7554–7561.

Hai Leong Chieu and Hwee Tou Ng. 2002. Named entity recognition. *Stanford Lecture CS229*, pages 1–7.

Sam Davidson, Jordan Hosier, Yu Zhou, and Vijay K. Gurbani. 2021. Improved Named Entity Recognition for Noisy Call Center Transcripts. In *W-NUT 2021 - 7th Workshop on Noisy User-Generated Text, Proceedings of the Conference*, pages 361–370.

Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL HLT - Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 4171–4186.

Nirav Diwan, Devansh Batra, and Ganesh Bagler. 2020. A named entity based approach to model recipes. *Proceedings - 36th International Conference on Data Engineering Workshops, ICDEW*, pages 88–93.

Tome Eftimov, Barbara Koroušić Seljak, and Peter Korošec. 2017. A rule-based named-entity recognition method for knowledge extraction of evidence-based dietary recommendations. *PLoS ONE*, 12(6):e0179488.

Oren Etzioni, Michael Cafarella, Doug Downey, Ana Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S Weld, and Alexander Yates. 2005. Unsupervised named-entity extraction from the Web: An experimental study. *Artificial Intelligence*, 165(1):91–134.

Qin Fang, Yane Li, Hailin Feng, and Yaoping Ruan. 2023. Chinese Named Entity Recognition Model Based on Multi-Task Learning. *Applied Sciences*, 13(8):4770.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by Gibbs sampling. In *ACL-05 - 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, pages 363–370, Ann Arbor, Michigan. Association for Computational Linguistics.

Mansi Goel and Ganesh Bagler. 2022. Computational gastronomy: A data science approach to food. *Journal of Biosciences*, 47(1):1–10.

Jiafeng Guo, Gu Xu, Xueqi Cheng, and Hang Li. 2009. Named entity recognition in query. In *Proceedings - 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009*, pages 267–274.

Md Jubaer Hossain, Mohammed Imamul Hassan Bhuiyan, and Zaowad Rahabin Abdullah. 2022. CpG Island Detection Using Transformer Model with Conditional Random Field. In *IBSSC 2022 - IEEE Bombay Section Signature Conference*, pages 1–5.

Bin Ji. 2023. VicunaNER: Zero/Few-shot Named Entity Recognition using Vicuna. *arXiv*.

Jushaan Kalra, Devansh Batra, Nirav Diwan, and Ganesh Bagler. 2020. Nutritional profile estimation in cooking recipes. In *36th International Conference on Data Engineering Workshops, ICDEW*, pages 82–87. IEEE.

Aman Kumar, Binil Starly, and Collin Lynch. 2023. ManuBERT: A pretrained Manufacturing science language representation model. *SSRN*.

Murari Kumar. 2023. An Algorithm for Automatic Text Annotation for Named Entity Recognition using spaCy Framework. *Research Square*, pages 1–18.

John Lafferty, Andrew Mccallum, and Fernando Pereira. 2001. Conditional Random Fields : Probabilistic Models for Segmenting and Labeling Sequence Data Abstract. *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289.

Wenzhong Liu and Xiaohui Cui. 2023. Improving Named Entity Recognition for Social Media with Data Augmentation. *Applied Sciences*, 13(9):5360.

Gang Luo, Xiaojiang Huang, Chin Yew Lin, and Zaiqing Nie. 2015. Joint named entity recognition and disambiguation. In *Conference on Empirical Methods in Natural Language Processing*, pages 879–888.

Bruno Mathis. 2022. Extracting Proceedings Data from Court Cases with Machine Learning. *Stats*, 5(4):1305–1320.

Honnibal Matthew, Ines Montani, Sofie Van Landeghem, and Boyd Adriane. 2020. spaCy Industrial-strength Natural Language Processing in Python.

Diego Mollá, Menno van Zaanen, and Daniel Smith. 2006. Named Entity Recognition for Question Answering. In *Proceedings ALTW*, pages 51–58.

Alexandre Passos, Vineet Kumar, and Andrew McCallum. 2014. Lexicon infused phrase embeddings for named entity resolution. *CoNLL 2014 - 18th Conference on Computational Natural Language Learning, Proceedings*, pages 78–86.

Dhrubajyoti Pathak, Sukumar Nandi, and Priyankoo Sarmah. 2022. AsNER - Annotated Dataset and Baseline for Assamese Named Entity recognition.

Nita Patil, Ajay Patil, and B V Pawar. 2020. Named Entity Recognition using Conditional Random Fields. *Procedia Computer Science*, 167:1181–1188.

Chantal Pellegrini, Ege Özsoy, Monika Wintergerst, and Georg Groh. 2021. Exploiting food embeddings for ingredient substitution. In *HEALTHINF 2021 - 14th International Conference on Health Informatics; Part of the 14th International Joint Conference on Biomedical Engineering Systems and Technologies, BIOSTEC 2021*, pages 67–77.

Desislava Petkova and W. Bruce Croft. 2007. Proximity-based document representation for named entity retrieval. In *International Conference on Information and Knowledge Management, Proceedings*, pages 731–740.

Gorjan Popovski, Stefan Kochev, Barbara Koroušić Seljak, and Tome Eftimov. 2019. Foodie: A rule-based named-entity recognition method for food information extraction. *ICPRAM 2019 - Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods*, 12:915–922.

Xiaoye Qu, Jun Zeng, Daizong Liu, Zhefeng Wang, Baoxing Huai, and Pan Zhou. 2023. Distantly-Supervised Named Entity Recognition with Adaptive Teacher Learning and Fine-Grained Student Ensemble. *Proceedings of the 37th AAAI Conference on Artificial Intelligence*, 37:13501–13509.

L. R. Rabiner and B. H. Juang. 1986. An Introduction to Hidden Markov Models. *IEEE ASSP Magazine*, 3(1):4–16.

Cosmina Radu, Carla Elena Staicu, Livia Maria Mitrică, Mihaela Dînsoreanu, Rodica Potolea, and Camelia Lemnaru. 2022. Extracting Settings from Multilingual Recipes with Various Sequence Tagging Models: an Experimental Study. In *Proceedings - 18th International Conference on Intelligent Computer Communication and Processing Conference, ICCP 2022*, pages 65–72.

Fillipe Barros Rodrigues, William Ferreira Giozza, Robson de Oliveira Albuquerque, and Luis Javier Garcia Villalba. 2022. Natural Language Processing Applied to Forensics Information Extraction With Transformers and Graph Visualization. *IEEE Transactions on Computational Social Systems*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT.

Motoki Sato, Hiroyuki Shindo, Ikuya Yamada, and Yuji Matsumoto. 2017. Segment-level neural conditional random fields for named entity recognition. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, pages 97–102.

Swardiantara Silalahi, Tohari Ahmad, and Hudan Studiawan. 2022. Named Entity Recognition for Drone Forensic Using BERT and DistilBERT. In *2022 International Conference on Data Science and Its Applications, ICoDSA 2022*, pages 53–58.

Muhammad Suleman, Muhammad Asif, Tayyab Zamir, Ayaz Mehmood, Jebran Khan, Nasir Ahmad, and Kashif Ahmad. 2022. Floods Relevancy and Identification of Location from Twitter Posts using NLP Techniques.

Muzamil Hussain Syed and Sun Tae Chung. 2021. Menuner: Domain-adapted bert based ner approach for a domain with limited dataset and its application to food menu domain. *Applied Sciences*, 11(13):6007.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023. GPT-NER: Named Entity Recognition via Large Language Models. *arXiv*.

Qikang Wei, Tao Chen, Ruifeng Xu, Yulan He, and Lin Gui. 2016. Disease named entity recognition by combining conditional random fields and bidirectional recurrent neural networks. *Database : the journal of biological databases and curation*, 2016:baw140.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang, and Wenjuan Han. 2023. Zero-Shot Information Extraction via Chatting with ChatGPT. *arXiv*.

Xiaoran Yang and Wenkang Huang. 2018. A conditional random fields approach to clinical name entity recognition. In *CEUR Workshop Proceedings*, volume 2242, pages 1–6.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2020. ErniE: Enhanced language representation with informative entities. *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, pages 1441–1451.

A. NER Tagging Inference Results

A.1. Error Analysis Comparing spaCy-transformer with Stanford NER

The two most frequent error patterns of Stanford NER that emerged in our analysis were the misclassification of the entity STATE as NAME and the entity UNIT as OTHER (see Figure 7). We exemplify these error patterns, showcasing instances where Stanford NER and spaCy-transformer differ in their predictions.

This study reveals a significant breakthrough: spaCy-transformer outperforms the established Stanford NER tagger in recipe entity classification tasks.

A.2. Erroneous Predictions using spaCy-transformer

The spaCy-transformer model exhibits erroneous predictions, which include misclassification of ingredient names and brands (see Figure 8).

B. Cleaning Protocols for Machine Annotated Dataset

- The model could not fully capture the unique culinary language dynamics different from our usual natural language. Color is an adjective, but it might be part of the ingredient. For example, 'Yellow lentils' where 'yellow' in natural language is a color and a usual natural language model would classify it as a 'STATE' of an ingredient. Other examples are red Romano pepper, red chilies, etc.
- Fixing the incorrect placement of named entities. These included the quantity incorrectly labeled as a unit or vice-versa and an ingredient incorrectly classified as a unit or vice-versa. A null value was used to indicate the absence of the unit in the unique list of training datasets.
- Append the fraction and integer together in the quantity as a string to avoid misclassification.
- Removal of special characters. For example, in "+1 chikoo", '+' is dropped.

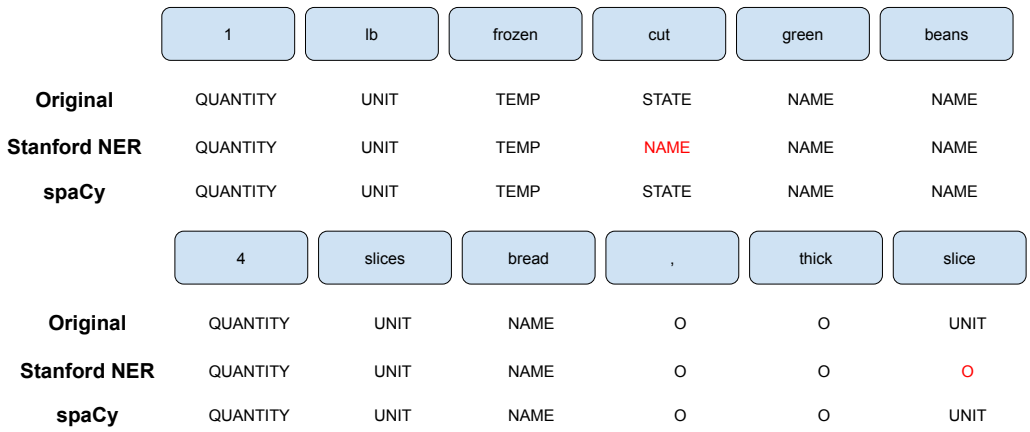

Figure 7: Error Analysis of Stanford NER tagger. Stanford NER tagger incorrectly classifies “cut” as NAME instead of STATE, which was correctly identified by spaCy-transformer. Similarly, “slice” classifies as OTHER instead of UNIT, which was correctly identified by spaCy-transformer.

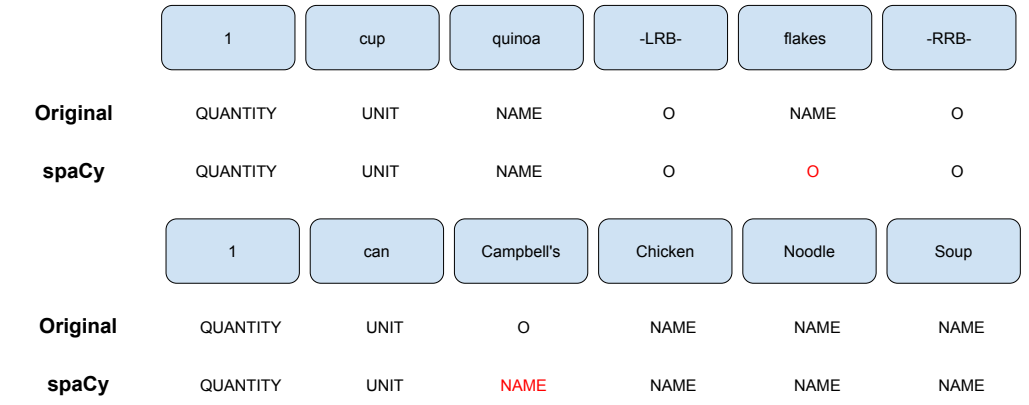

Figure 8: Error Analysis of spaCy-transformer. spaCy misclassifies “flakes" as OTHER (O) instead of a specific form of quinoa (NAME). Similarly, “Campbell’s" is the chicken noodle soup brand name (O) instead of an ingredient name (NAME). -LRB- and -RRB- stand for left and right round brackets, respectively.