Semi-supervised Named Entity Recognition to solve label scarcity challenges for E-Commerce use-cases

Ayesha Siddiqa*
Flipkart
Bangalore, India
midthur.as@flipkart.com

Suman Banerjee*
Flipkart
Bangalore, India
suman.banerjee@flipkart.com

Nikesh Garera
Flipkart
Bangalore, India
nikesh.garera@flipkart.com

ABSTRACT

The E-commerce market is dynamically growing and it has become essential for the industry to build top-notch search and customer service natural language understanding systems and platforms. Named entity recognition (NER) is a critical component of such natural language understanding systems. NER is required to identify the entity types that can help search systems retrieve relevant products, create knowledge graphs that are essential for catalog population and completion, etc. All these components are put together to offer an appealing and seamless shopping experience for the customers. However, building effective NER solutions on the user-generated text is a challenging task due to ambiguity or variations of the entity mentions, dearth of labeled data for training ML models and noisy or incoherent annotations in the available labeled data. In this paper, we propose three end-to-end solutions to address these challenges - character and sub-word based embedding methods for an effective NER architecture, a self-training based approach and noisy labels handling approach. We highlight the effectiveness of our proposed solutions by doing empirical evaluations on the three use-cases to which we have catered, namely Voice Search, Text Search and Buying Assistant. We achieved relative improvements of 15.50%, 3.42%, and 8.48% in the weighted F_1 scores for these three datasets in comparison to baselines.

CCS CONCEPTS

• **Named entity recognition**; • **Semi supervised learning**; • **Noisy labels**;

ACM Reference Format:
Ayesha Siddiqa, Suman Banerjee, and Nikesh Garera. 2022. Semi-supervised Named Entity Recognition to solve label scarcity challenges for E-Commerce use-cases. In *Content Understanding and Generation for E-commerce Workshop (EcomGen, KDD).* ACM, Washington, DC, USA, 8 pages.

1 INTRODUCTION

Named Entity Recognition (NER) aims at extracting relevant entities from unstructured text and assigning them to respective entity types. It is one of the fundamental building blocks in a natural language understanding pipeline. In the e-commerce domain, it contributes to a number of downstream applications such as: extracting the entities in search user queries for the retrieval of relevant products[35], identification of entities in a Question Answering system[32], Intent Classification[41], creation of a Knowledge Graph[1] from unstructured product descriptions, *etc.* NER on user-generated text is a challenging task because of the variations with which users can mention the entity names. A NER system can encounter out of vocabulary (OOV) words, which makes it difficult for the models to predict the correct entity types. For example, users often mention new brand names in their search queries with either spell errors or words unavailable in the training data. In order to handle OOV words in our NER models, we use character and sub-word based embeddings.

Current neural network based NER models have demonstrated state-of-the-art performance on the NER [44] task. However, their efficacy is dependent on the availability of huge amounts of clean labeled data. Annotation of such large accurate datasets is expensive, time-consuming and requires domain expertise. On the other hand, unlabeled queries are available in abundance. This facilitates the need to incorporate Semi-Supervised Learning (SSL)[25] based NER solutions in e-commerce, thereby mitigating the problem of labeled data shortage. We tackle this problem using a special kind of SSL approach called *Self-Training*[18]. This self-training approach is built on the teacher-student framework, where a teacher model is trained on the labeled dataset. The teacher model is used to propagate pseudo-labels to the unlabeled data. After this step, student model is trained with both labeled and pseudo-labeled data and the process is repeated till a predefined number of steps. However, a major drawback of self-training approaches is that they are quite sensitive to label noise or label inconsistencies.

Live user queries from the production pipeline are collected and annotated periodically. The deployed NER system needs to adapt to the language variations over time and thus the annotation guidelines keep evolving. Moreover, queries are annotated in batches by different groups of annotators. This introduces a lot of noisy and inconsistent labels [10]. Training deep neural networks with such noisy labels leads to degraded performance because the standard cross-entropy loss overfits on these noisy labels. A modified version of the cross-entropy loss was proposed in [33] for the image classification task. In this work, we adapt that modification of cross-entropy loss in sequence labeling tasks to handle such overfitting to label noise.

We evaluate our approaches on three use-cases: (i) Voice Search (ii) Text Search and (iii) Buying Assistant (BA). Voice search is used by our customers to search for products using speech as the input modality. Text search is the primary tool used by our customers

*Both authors contributed equally to this work

EcomGen, KDD, 2022, Washington DC, USA

to search products using unstructured text as the input. Buying Assistant is a new feature that helps our customers with answers to their questions before finalizing the purchase decision. We describe our experiments and evaluate our basic models, self-training approaches and label-noise mitigation technique on test sets comprised of user queries from all three use-cases. We have achieved weighted F_1 scores of 83.15%, 87.96% and 85.91% on Voice Search, Text Search and BA respectively.

Our major contributions in this paper are:

- We build NER solutions with OOV handling methods such as character level CNN, fasttext and BPE on three use-cases - Text search, Voice search and Buying Assistant.
- We further propose a teacher-student self-training framework to mitigate dependency on labeled data by utilizing pseudo-labels on unlabeled data.
- Finally, we adapt a modification of CE loss to handle label noise in the training data.
- We obtained relative improvements of 15.50%, 3.42% and 8.48% in the weighted F_1 scores for Voice Search, Text Search and BA datasets respectively.

2 RELATED WORK

In this section, we will briefly explain the previous work done for Named Entity Recognition (NER) problem in supervised and semi-supervised settings. Early solutions proposed for NER are knowledge based approaches which rely on lexical clues available in the queries. While these approaches are easier to interpret, their coverage is low and cannot generalize to new domains. As a result, these approaches have high *precision* and low *recall*. Feature engineering based solutions rely on hand crafted features encoded as feature vectors. [43] were the first ones to come up with a robust feature engineering based approach utilizing orthographic features like capitalization, alphabets, etc., along with trigger words, person names and built a sequence tagger using Hidden Markov Model (HMM). It is a tedious task to come up with features for the NER problem settings where the number of domains is huge and features from one domain may not be not reusable for another domain.

To overcome the drawbacks of feature based approaches deep learning based end-to-end solutions were proposed. These solutions replace hand crafted features with learnable feature vectors called embeddings and use them to train NER models. These embeddings can be of two types: word based [17] or character based [19]. Further, there could be hybrid approaches similar to [21] which utilizes both character and word-level features with BiLSTM-CRF as their backbone architecture to effectively recognize entities. These approaches help us in building robust NER models however, they require large human annotated data which is time consuming and expensive to obtain.

To mitigate the dependency on labeled data and to effectively utilize unlabeled data, Semi-Supervised Learning (SSL) approaches come to our rescue. Among all the SSL approaches, a specific SSL approach called Self-Training [18] has been recently demonstrated to give substantial gains in tasks like Text Classification [24], Machine Translation [15], slot filling [34]. Self-training is used to propagate pseudo-labels to unlabeled data which helps us to train a robust model combining human labeled and pseudo-labeled data. However, for this approach to be successful we have to do an intelligent selection of pseudo labeled samples. To this end, we have explored various sample uncertainty measures and adopted the most intuitive one i.e., Bayesian Active Learning by Disagreement (BALD) proposed by [16] for the NER task.

Numerous approaches have been explored in the past for handling noisy labels in the training of Deep Neural Networks (DNN) [30]. Some studies [2, 4, 12] proposed architectural changes such as a noise adaptation layer on top of the softmax layer to learn the noise transition matrix, whereas others [14, 37, 38] came up with dedicated architectures to handle more complex and realistic noise. Regularization based approaches such as loss-based gradient clipping [23] and *robust early learning* [36] prevent the models from overfitting to the label noise. The regularization can be implicitly obtained through adversarial training [13], label smoothing [22] or linear interpolation of noisy examples in the embedding space [40]. Our adaptation of SCE loss in NER is more closely related to the family of noise handling methods through designing robust loss functions. Mean Absolute Error (MAE) has proven to be noise-tolerant [11], but it is difficult to train DNNs with MAE. Generalized Cross-Entropy (GCE) [42] was proposed to combat this problem by combining the advantages of CE and MAE in a single formulation. Symmetric Cross Entropy (SCE) [33] was developed from the perspective of information theory, wherein if the true label distribution is noisy then we need to consider the reverse direction of KL-Divergence for penalizing noisy labels. Therefore the authors added a reverse cross entropy term to CE to make it robust. SCE was introduced for vision tasks and in this paper, we adapt the formulation in our NER task.

3 PROBLEM DEFINITION

The training corpus can be defined as $\mathcal{D}_L = \{(X^i, Y^i)\}_{i=1}^N$, where each $X^i = x_1, x_2, ..., x_n$ represents a sentence with n tokens and each $Y^i = y_1, y_2, ..., y_n$ represents the entity types in the label sequence. Traditionally, NER is defined as a supervised sequence labelling problem wherein the task is to classify each input token x_j into one of K entity types. For example, for an input sentence: "*show me navy blue nike shoes*", the model needs to produce the following entity types corresponding to each token: "*O O B-color I-color B-brand B-category*". Here the model predicts the entity boundaries corresponding to each entity type using the BIO format, wherein the start token of an entity mention is specified by "*B-type*", the intermediate tokens by "*I-type*" and "*O*" specifies non-entity tokens. In this work, we use the BIO format only for the Buying Assistant and use the above mentioned formulation without the boundary tags in Voice and Text Search use-cases.

In our semi-supervised problem setup, we have a set of M ($M >> N$) unlabeled sentences $\mathcal{D}_U = \{X^i\}_{i=1}^M$ in addition to $\mathcal{D}_L$ where M stands for number of unlabeled samples and N stands for number labeled samples. Our objective is to utilize these unlabeled samples in order to tackle the problem of labeled data scarcity. Due to evolving annotation guidelines and differences in batches of annotators, we may encounter noisy labels in $\mathcal{D}_L$. We may also encounter noisy pseudo-labels, which are being generated by a teacher model

trained on $\mathcal{D}_L$. This poses an additional challenge while designing an NER model which is resilient to such label noises.

4 PROPOSED APPROACHES

In this section, we describe our base NER model, our semi-supervised solution to the label scarcity problem and a modification to the loss function that makes the model robust to label noise.

4.1 Basic Architecture

We now explain our base model and describe the character and sub-word features used to handle the OOV problem.

4.1.1 Base Model: Let the word embeddings corresponding to the input tokens $x_1, x_2, ..., x_n$ be denoted by $\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_n}$. We experiment with randomly initialized word embeddings as well as GloVe [26] embeddings. We model the sequence labelling task with a bidirectional LSTM (BiLSTM)[28] and a decoding layer on top of it. Specifically, the sequential dependencies are captured in the hidden states $\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_n$ of the BiLSTM:

$$\begin{aligned} \mathbf{h}_i^f &= BiLSTM_f(\mathbf{x}_i, \mathbf{h}_{i-1}^f) \\ \mathbf{h}_i^b &= BiLSTM_b(\mathbf{x}_i, \mathbf{h}_{i-1}^b) \\ \mathbf{h}_i &= \mathbf{h}_n^f \oplus \mathbf{h}_1^b \end{aligned} \tag{1}$$

where the $\oplus$ symbol denotes concatenation. We use a decoding layer on top of the BiLSTM to obtain the entity type predictions. Specifically, we use a feedforward layer followed by an output softmax layer as the decoder.

4.1.2 Character and Subword Features: To handle OOV words, we experimented with a character-level CNN. Specifically, we used windows of sizes $D \times W$ with $W = 1, 2, 3$ and 4, on character embeddings (of D dimensions) along the length of the word. The extracted features for each window size are concatenated to obtain the final character CNN embeddings ($\mathbf{d}_i$) for each token. Finally, the character CNN embeddings are concatenated with the word embeddings ($\mathbf{x}_i \oplus \mathbf{d}_i$) to form the input to the BiLSTM. For OOV words, such character based embeddings provide some distinguishing features in comparison to just using the word embedding for *_UNK_* token.

We also experimented with Fasttext [3] features as the character level representations. Fasttext incorporates the morphology of words into the word representations and therefore can help in cases where the OOV word is a morphological variant of a word that is present in the vocabulary. To obtain Fasttext vectors specific to our domain, we pretrained them using our own corpora instead of using the already available ones. We also experimented with Byte-pair Encoding (BPE) [29] as the sub-word features for the OOV problem. BPE is a data compression technique [8] adapted to the task of subword segmentation, where it merges the most frequent character sequences into a single subword iteratively.

4.2 Self-Training

To minimize annotation bottleneck we have propose an approach that utilizes a small amount of labeled data and a huge amount of unlabeled data called *Self-Training*. In this self-training framework, we have a *teacher* model which is trained on the small amount of labeled data and is used to propagate labels to the unlabeled data available to us. Since these labels are not obtained with manual annotation they are called *pseudo-labels*. Labeled data and pseudo labeled data combined are used to train the *student* model. Teacher-student training is repeated alternately for N steps or till convergence. In general scenarios, the student model will have the same architecture and configuration as the teacher model.

For self-training approaches to be successful we must have an intelligent sample selection of pseudo labeled samples for training the student model. Few intuitive sampling strategies to consider are include only those samples where the teacher model is highly confident (likely easy samples) or those where the teacher model is least confident (likely hard samples). These strategies have their own drawbacks, sampling only from the easy set for student training leads to minor gains utilizing the self-training approach. This is because the model has already learned these patterns from the labeled data. On the contrary, sampling only from the hard set leads to noisy pseudo-labeled samples being selected and gradual drifts from the initial labeled set of ground truth examples. Thus, We need a sampling strategy that judiciously selects samples where the model is uncertain and at the same it does not select samples far away from the initial labeled set to avoid drifts. For this judicious sampling, we adopted *Bayesian Active Learning by Disagreement (BALD)* measure proposed by [16] given by the formula

$$BALD = H[p(Y|X, \mathcal{D})] - E_{p(\theta|\mathcal{D})} H[p(Y|X, \theta)] \tag{2}$$

where X are inputs, Y are outputs, $\mathcal{D}$ is the dataset, H is the entropy function and θ are model parameters. The first term in the above equation looks for an input X for which the model is highly uncertain about output Y. In other words, the output has high marginal entropy. The second term looks for a data point with low expected conditional uncertainty. The equation can be interpreted as seeking the X for which the model parameters under the posterior make confident predictions, but these predictions are highly diverse. That is, the parameters disagree about the output Y, hence we name this formulation Bayesian Active Learning by Disagreement ie., higher the model uncertainty of a sample higher is its BALD value.

[9] hypothesized that approximate BALD scores can be computed using Monte Carlo (MC) Dropout. Calculation of individual token BALD scores using MC dropout requires multiple stochastic passes, and in each of them, we have to perform inference of the model. The BALD score for the entire sample or query can be obtained by taking an average of individual token BALD scores. While we intuitively explain BALD approximation we point interested readers to [9] for the mathematical derivations of how BALD scores can be computed using MC dropout and do not include them here for brevity. For our commonly used sequences taggers like BiLSTM-CRF taggers, MC dropout can be applied in the following ways: (1) MC word dropout i.e., randomly drop the entire after the word embedding layer, (2) MC locked dropout i.e., drop the same neurons in the embedding space of a recurrent layer for a whole sequence and (3) MC all i.e., both MC word and MC locked incorporated in the model. We finally select samples whose BALD scores are <= 0.5 for student training.

4.3 Label Noise mitigation

Neural networks for classification are typically trained with Cross-entropy (CE) loss. However, CE drives the training of networks

Table 1: Dataset Statistics

	Voice Search	Text Search	BA bot
Train	13,16,419	6,58,030	60,359
Test	4,239	39,045	6,721
unlabeled	51,06,065	14,00,124	1,37,974
Pseudo labeled (Best teacher)	31,50,000	8,90,000	84,956
Vocab Size	57,928	46,134	24,065
Unique words in Test	2,837	7,715	5,487
OOV words in Test	90	1,414	1,830

Table 2: Basic architecture parameters

Hyperparameter	Voice Search	Text Search	BA bot
Best Embedding method	Fasttext	GloVe	Fasttext
Word Embedding dim.	512	512	300
Char Embedding dim.	300	300	300
BiLSTM layers	3	2	2
BiLSTM nodes	640	128	128
Feedforward Hidden nodes	800	800	800

in a class-biased manner [33, 39]. The models tend to converge on some of the classes faster (easy classes) while it takes longer to converge on the other (hard classes). In the presence of label noise, this problem becomes even more pronounced wherein the model overfits the label noise of easy classes and learns in the hard classes. To mitigate this problem, the authors in [33] proposed to add another term to CE loss, inspired by the symmetric version of KL-divergence (KLD).

Let the predicted distribution of the model be denoted by $p(x)$ and the true distribution by $q(x)$. Typically, $q(x)$ is one-hot encoded and therefore minimizing CE will be equivalent to minimizing $KLD(q||p)$. It represents the penalty of encoding samples from $q(x)$ when using a code optimized for $p(x)$. If $q(x)$ is noisy then we also need to consider the reverse $KLD(p||q)$ to penalize encoding samples from $p(x)$ using a code for $q(x)$. Thus, the authors add a reverse cross entropy (RCE) term to CE to formulate symmetric cross-entropy (SCE):

$$\begin{aligned} SCE &= \alpha CE + \beta RCE = \alpha H(q,p) + \beta H(p,q) \\ RCE &= -\sum_{k=1}^{K} p(x_k) \log q(x_k) \end{aligned} \tag{3}$$

Here, α and β are mixing hyper-parameters to control the trade-off between the noise tolerance of RCE and convergence of CE. Since $q(x_k)$ is inside the logarithm, it can cause numerical instabilities with $q(x)$ being one-hot. Therefore, we define $\log 0 = A$ (where $A < 0$ is some constant).

5 EXPERIMENTAL SETUP

We perform exhaustive experimentation for our proposed approaches with different architectures, embeddings and loss functions. We use weighted Precision, Recall and F_1 scores as the evaluation metrics. Since the number of entities in our datasets is highly imbalanced we also report macro F_1 scores in table 5.

5.1 Datasets

We have evaluated our proposed approaches on three datasets obtained from our internal Voice Search, Text Search and BA user queries. We present a summary and statistics of labeled and unlabeled data sets available with us in the table 1. Voice search queries are a combination of Hindi, English and Hinglish (i.e., both Hindi and English) languages. We have 93 entity tags in Voice search like *idealFor*, *category*, *brand*, etc. User queries from the voice search product were collected and annotated. Our second dataset is from the text search domain. These queries are predominantly in English with a few of them in Hindi and Hinglish. Unlike Voice search, this dataset is limited only to the Lifestyle domain. We have 127 entity-tags for the text search dataset. Few examples of them are *Color*, *Category*, *Fabric* etc. Live text search queries were sampled in a stratified manner based on impressions and were annotated to create this dataset. Our last dataset is obtained from the customer interactions with Buying Assistant (BA). These queries are predominantly in English with few in Hindi and Hinglish. BA use-case is has 23 entity classes and Unlike other use-cases, BA NER caters to both generic entity tags like *PaymentType*, *BankName* etc., and Lifestyle specific entities like *FootwearLength*, *Height*, *Weight* etc.

5.2 Basic Architecture

Our basic architecture is adopted from the neural model for Named Entity Recognition proposed by [21]. We chose the number of BiLSTM nodes, BiLSTM layers and feedforward layer nodes by hyper-parameter tuning on a validation set. We summarize these choices in table 2. We use a dropout [31] of 0.3 for regularization and Adam [20] optimizer with an initial learning rate of 0.001 for training our models. To mitigate the domain gap, instead of using the publicly available word embeddings for GloVe and Fasttext, we pre-trained these embeddings on our datasets. We also compared our results with the Spacy NER model as a baseline.[1]

5.3 Self-Training

For the self-training experiments, we use best teacher models from table 3 and utilize unlabeled data available with us. As can be observed from the table 3, best teacher NER models for Voice search, text search and BA are BiLSTM+CharCNN+Fasttetxt, BiLSTM+CharCNN+Glove and BiLSTM+CharCNN+Fasttext respectively. These best teacher models and their SCE variants are used to infer pseudo labels for the unlabeled data available to us (Refer to the table 1 for the number of unlabeled samples). For each of the samples, during the inference phase, a BALD score for the entire sample is computed using equation (2). If the BALD score is <= 0.5 then the pseudo labeled sample will be used for student model training otherwise discarded.

5.4 Label Noise mitigation

The Reverse Cross Entropy (RCE) term is mixed with Cross Entropy (CE) term in the loss with mixing hyper-parameters α and β. For the Best Teacher + SCE model in (i) Voice Search we used $\alpha = 2$ and $\beta = 1$, (ii) Text Search we used $\alpha = 5$ and $\beta = 7$ and in (iii) BA we used $\alpha = 25$ and $\beta = 1$. These values for α and β were chosen by

[1] https://spacy.io/usage/training

Table 3: Weighted Precision, Recall and F_1 scores of baselines with character and subword embeddings

Models	Voice Search			Text Search			BA bot		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Spacy	73.11	70.91	71.99	85.73	84.39	85.05	**86.83**	82.02	84.36
BiLSTM	83.75	80.65	82.17	87.60	86.33	86.96	80.92	77.53	79.19
BiLSTM+charCNN	83.73	80.94	82.31	87.96	86.77	87.36	82.01	87.26	84.55
BiLSTM+charCNN+GloVe	84.23	81.38	82.78	**88.13**	86.80	**87.47**	82.72	**87.91**	85.24
BiLSTM+charCNN+BPE	83.94	81.39	82.65	88.01	86.92	87.46	79.58	84.82	82.12
BiLSTM+charCNN+Fasttext	**84.39**	**81.92**	**83.14**	87.90	**86.96**	87.43	83.57	87.82	**85.64**

Table 4: Weighted Precision, Recall and F_1 scores of Self-training and Noise-handling (SCE) methods

Models	Voice Search			Text Search			BA bot		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Best Teacher	84.39	81.92	83.14	88.13	86.80	87.46	83.57	87.82	85.64
Best Teacher + SCE	84.32	81.70	82.99	88.05	87.14	87.59	**83.93**	**87.98**	**85.91**
Self-Training Student model	**84.83**	**81.53**	**83.15**	88.27	86.88	87.57	81.87	87.45	84.57
Self-Training Student model + SCE	84.06	80.74	82.37	**88.67**	**87.27**	**87.96**	82.62	86.77	84.64

Table 5: Macro Precision, Recall and F_1 scores of Self-training and Noise-handling (SCE) methods

Models	Voice Search			Text Search			BA bot		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Best Teacher	64.32	59.12	61.61	**56.90**	54.95	55.91	56.37	**58.98**	**57.65**
Best Teacher + SCE	65.96	59.96	62.82	53.76	52.04	52.89	54.82	56.67	55.73
Self-Training Student model	**68.57**	**63.64**	**66.01**	55.61	**56.63**	**56.12**	**56.40**	52.37	54.31
Self-Training Student model + SCE	64.07	58.12	60.95	53.59	52.38	52.98	55.56	54.46	55.00

hyper-parameter tuning. The same values of α and β were retained for the respective datasets while fine-tuning SCE student models.

6 RESULTS AND INSIGHTS

In this section, we explain the results obtained from our extensive experiments and some insights into the model behaviors.

6.1 Results

The weighted Precision, Recall and F_1 scores of our basic architecture with various embedding methods are summarized in table 3. All our models outperform the Spacy baseline. For all the three datasets, character, n-gram and subword based embeddings help in improving the model's performance in comparison with the basic BiLSTM model. In the absence of pseudo-supervision, the best F_1 scores are obtained using Fasttext in Voice Search (83.14%) and BA (85.64%) and using GloVe in Text Search (87.47%). The BA test set has the highest percentage of OOV words (33.35% in table 1) and therefore the benefits from using character and n-gram based embeddings are the highest in this dataset compared to others. These results show the benefits of using character and pre-trained embeddings for tackling the OOV problem in NER.

The weighted precision, recall and F_1 scores of self-training along with SCE loss have been summarized in table 4. Due to the usual class imbalance in the datasets, we also report the macro F_1 scores in table 5. In Voice Search, the best weighted F_1 score (83.15) is obtained by self-training with the student model using CE loss. Even though we have minor improvement in table 4, the macro scores in table 5 show a large improvement while using self-training. This shows that self-training with unlabeled data helps in improving the scores of minority labels, for which the labeled data are scarce. The addition of SCE loss in Voice Search did not help. In Text Search, self-training and SCE loss contributed to the improvements in weighted scores and the best F_1 score (87.96%) is obtained by a combination of both. For macro results, the best F_1 score (56.12%) was obtained by self-training without SCE. In the BA dataset, self-training did not provide benefits and the best model was obtained by using SCE loss with best teacher from table 3. This is because the size of unlabelled data was small and the confidence of teacher model was low on the pseudo labeled samples.

6.2 Interpretable attributes

The changes in model behavior under different conditions are not reflected by a single F_1 score, and therefore identifying the strengths and weaknesses of these models remains an open problem. To combat this problem, [7] introduced a method of interpretable evaluation by splitting test sets into different buckets according to a few characteristic attributes of queries. Similarly, we have computed the following attributes of queries in test sets:

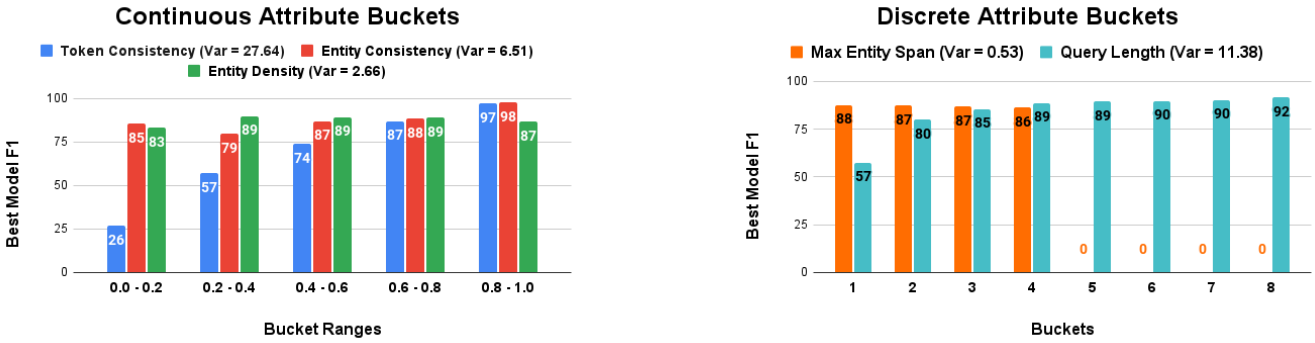

Figure 1: Best Model F1 scores of Text Search on interpretable test buckets

- **Maximum entity span**: The maximum number of tokens in a single entity mentioned.
- **Entity density**: Number of entity tokens in a query, divided by the total number of tokens in that query.
- **Entity consistency**: The number of occurrences of an entity mentioned in the train set with a specific label sequence, divided by the total number of occurrences of that entity.
- **Token consistency**: Number of occurrences of a token in the train set with a specific label, divided by the total number of occurrences of that token.
- **Query Length**: The number of tokens in a query.

After computing these attribute values, we group the test queries into equally spaced buckets according to the attribute values. Note that the consistency values are undefined for OOV words and therefore we exclude queries with OOV words for computing the entity and token consistencies. We compute the weighted F_1 scores for each of these buckets and the standard deviation of these scores across all buckets. The results are summarized in figure 1. This evaluation is performed only for the Text Search test set because it is the only dataset with a large enough size to be split into buckets with decent sizes. The results show that entity span length is not a problem for our NER models, as the F_1 scores do not vary much.

We see that the F_1 scores first increase and then decrease as the entity density increases. This shows that our NER models perform poorly in queries with high entity density. But the standard deviation values show that our models struggle the most in queries with low label consistency. Therefore tokens and entities which different label distributions in train and test are a major problem for our models. The F_1 scores reach as high as 97 for buckets containing queries with high label consistencies. This shows that ensuring high label consistencies through better tagging instructions and the removal of incorrect labels should provide huge benefits. The results also show that queries with short lengths are a problem for our NER models as they lack enough context information for the models to predict correctly.

6.3 Impact of spell correction

Voice search and Text search datasets already have spell corrected queries, unlike BA queries. Therefore, we tried out the our in-house spell correction model on the BA test queries. We analyzed around 3k BA queries with and without spell error correction. We found that F_1 scores of our best model improved from **80.22**% to **82.42**% on the entity tokens. We further performed a qualitative analysis and found that spell correction helps a lot for the entity tokens belonging to the entity classes *brand*, *size* and *age*.

6.4 BERT based variants

Recent advances in LM pretraining based methods [5, 6, 27] show state-of-the-art results in public NER tasks. Along with the variants of BiLSTM baselines we worked on utilizing LM pretraining based methods. We tried finetuning of our in-house mBERT, BERT and XLM on Voice Search test set which showed F_1 scores of 81.32%, 82.32% and 81.83% respectively. Therefore, we did not obtain any benefit using such large models in Voice Search and a primary reason was short query lengths in user data (avg length $\sim$ 4). Further, we did not try these large models on Text Search and BA because of the strict latency constraints of $\sim 30ms$ in these products.

7 CONCLUSION AND FUTURE WORK

In this paper, we tackled the challenges of OOV words, label scarcity, and noisy labels, that are commonly observed in the real world NER use-cases. To alleviate, the problem of OOV words we relied on character, n-gram and subword based embeddings. We got maximum gains on the BA test set which had huge number of OOV words. For the problem of label scarcity, we use a self-training based approach which assesses quality of pseudo labels using an uncertainty measure (i.e., BALD). The student models trained using these pseudo-labels resulted in improved performance for Voice search and Text Search. Finally, to handle the incorrect signals from noisy labels, we adapt a robust loss function and observe benefits in the Text Search dataset. Overall, we achieved relative improvements of 15.50%, 3.42% and 8.48% in the weighted F_1 scores for Voice Search, Text Search and BA bot datasets respectively. In the future, this work can be improved with sophisticated pseudo label quality assessment functions and formulating robust loss functions with noise tolerance and faster convergence. Finally, we can work on improving the latencies of BERT based models through distillation or quantization, for reaping the benefits of these large models.

REFERENCES

[1] Tareq Al-Moslmi, Marc Gallofré Ocaña, Andreas L. Opdahl, and Csaba Veres. 2020. Named Entity Extraction for Knowledge Graphs: A Literature Overview. *IEEE Access* 8 (2020), 32862–32881. https://doi.org/10.1109/ACCESS.2020.2973928

[2] Alan Joseph Bekker and Jacob Goldberger. 2016. Training deep neural-networks based on unreliable labels. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2016, Shanghai, China, March 20-25, 2016*. IEEE, 2682–2686. https://doi.org/10.1109/ICASSP.2016.7472164

[3] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics* 5 (2017), 135–146. https://doi.org/10.1162/tacl_a_00051

[4] Xinlei Chen and Abhinav Gupta. 2015. Webly Supervised Learning of Convolutional Networks. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, 1431–1439. https://doi.org/10.1109/ICCV.2015.168

[5] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault (Eds.). Association for Computational Linguistics, 8440–8451. https://doi.org/10.18653/v1/2020.acl-main.747

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, Jill Burstein, Christy Doran, and Thamar Solorio (Eds.). Association for Computational Linguistics, 4171–4186. https://doi.org/10.18653/v1/n19-1423

[7] Jinlan Fu, Pengfei Liu, and Graham Neubig. 2020. Interpretable Multi-dataset Evaluation for Named Entity Recognition. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, 6058–6069. https://doi.org/10.18653/v1/2020.emnlp-main.489

[8] Philip Gage. 1994. A new algorithm for data compression. *The C Users Journal archive* 12 (1994), 23–38.

[9] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. 2017. Deep Bayesian Active Learning with Image Data. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017 (Proceedings of Machine Learning Research, Vol. 70)*, Doina Precup and Yee Whye Teh (Eds.). PMLR, 1183–1192. http://proceedings.mlr.press/v70/gal17a.html

[10] Jose Garrido Ramas, Giorgio Pessot, Abdalghani Abujabal, and Martin Rajman. 2021. Identifying and Resolving Annotation Changes for Natural Language Understanding. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Papers*. Association for Computational Linguistics, Online, 10–18. https://doi.org/10.18653/v1/2021.naacl-industry.2

[11] Aritra Ghosh, Himanshu Kumar, and P. S. Sastry. 2017. Robust Loss Functions under Label Noise for Deep Neural Networks. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, Satinder P. Singh and Shaul Markovitch (Eds.). AAAI Press, 1919–1925. http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14759

[12] Jacob Goldberger and Ehud Ben-Reuven. 2017. Training deep neural-networks using a noise adaptation layer. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net. https://openreview.net/forum?id=H12GRgcxg

[13] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and Harnessing Adversarial Examples. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1412.6572

[14] Bo Han, Jiangchao Yao, Gang Niu, Mingyuan Zhou, Ivor W. Tsang, Ya Zhang, and Masashi Sugiyama. 2018. Masking: A New Perspective of Noisy Supervision. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (Eds.). 5841–5851. https://proceedings.neurips.cc/paper/2018/hash/aee92f16efd522b9326c25cc3237ac15-Abstract.html

[15] Junxian He, Jiatao Gu, Jiajun Shen, and Marc'Aurelio Ranzato. 2020. Revisiting Self-Training for Neural Sequence Generation. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net. https://openreview.net/forum?id=SJgdnAVKDH

[16] Neil Houlsby, Ferenc Huszar, Zoubin Ghahramani, and Máté Lengyel. 2011. Bayesian Active Learning for Classification and Preference Learning. *CoRR* abs/1112.5745 (2011). arXiv:1112.5745 http://arxiv.org/abs/1112.5745

[17] Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF Models for Sequence Tagging. *CoRR* abs/1508.01991 (2015). arXiv:1508.01991 http://arxiv.org/abs/1508.01991

[18] H. J. Scudder III. 1965. Probability of error of some adaptive pattern-recognition machines. *IEEE Trans. Inf. Theory* 11, 3 (1965), 363–371. https://doi.org/10.1109/TIT.1965.1053799

[19] Yoon Kim, Yacine Jernite, David A. Sontag, and Alexander M. Rush. 2016. Character-Aware Neural Language Models. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, Dale Schuurmans and Michael P. Wellman (Eds.). AAAI Press, 2741–2749. http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12489

[20] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1412.6980

[21] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, San Diego, California, 260–270. https://doi.org/10.18653/v1/N16-1030

[22] Michal Lukasik, Srinadh Bhojanapalli, Aditya Krishna Menon, and Sanjiv Kumar. 2020. Does label smoothing mitigate label noise?. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 6448–6458. http://proceedings.mlr.press/v119/lukasik20a.html

[23] Aditya Krishna Menon, Ankit Singh Rawat, Sashank J. Reddi, and Sanjiv Kumar. 2020. Can gradient clipping mitigate label noise?. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net. https://openreview.net/forum?id=rklB76EKPr

[24] Subhabrata Mukherjee and Ahmed Hassan Awadallah. 2020. Uncertainty-aware Self-training for Few-shot Text Classification. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (Eds.). https://proceedings.neurips.cc/paper/2020/hash/f23d125da1e29e34c552f448610ff25f-Abstract.html

[25] Yassine Ouali, Céline Hudelot, and Myriam Tami. 2020. An Overview of Deep Semi-Supervised Learning. *CoRR* abs/2006.05278 (2020). arXiv:2006.05278 https://arxiv.org/abs/2006.05278

[26] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar, 1532–1543. https://doi.org/10.3115/v1/D14-1162

[27] Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How Multilingual is Multilingual BERT?. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, Anna Korhonen, David R. Traum, and Lluís Màrquez (Eds.). Association for Computational Linguistics, 4996–5001. https://doi.org/10.18653/v1/p19-1493

[28] M. Schuster and K.K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing* 45, 11 (1997), 2673–2681. https://doi.org/10.1109/78.650093

[29] Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Berlin, Germany, 1715–1725. https://doi.org/10.18653/v1/P16-1162

[30] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. 2020. Learning from noisy labels with deep neural networks: A survey. *arXiv preprint arXiv:2007.08199* (2020).

[31] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* 15, 1 (2014), 1929–1958. http://dl.acm.org/citation.cfm?id=2670313

[32] Antonio Toral, Elisa Noguera, Fernando Llopis, and Rafael Muñoz. 2005. Improving Question Answering Using Named Entity Recognition. In *Natural Language Processing and Information Systems, 10th International Conference on Applications of Natural Language to Information Systems, NLDB 2005, Alicante, Spain, June 15-17, 2005, Proceedings (Lecture Notes in Computer Science, Vol. 3513)*, Andrés Montoyo, Rafael Muñoz, and Elisabeth Métais (Eds.). Springer, 181–191. https://doi.org/10.1007/11428817_17

[33] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. 2019. Symmetric Cross Entropy for Robust Learning With Noisy Labels. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (2019), 322–330.

[34] Yaqing Wang, Subhabrata Mukherjee, Haoda Chu, Yuancheng Tu, Ming Wu, Jing Gao, and Ahmed Hassan Awadallah. 2021. Meta Self-training for Few-shot Neural Sequence Labeling. In *KDD '21: The 27th ACM SIGKDD Conference on*

Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021, Feida Zhu, Beng Chin Ooi, and Chunyan Miao (Eds.). ACM, 1737–1747. https://doi.org/10.1145/3447548.3467235

[35] Musen Wen, Deepak Kumar Vasthimal, Alan Lu, Tiantian Wang, and Aimin Guo. 2019. Building Large-Scale Deep Learning System for Entity Recognition in E-Commerce Search. *Proceedings of the 6th IEEE/ACM International Conference on Big Data Computing, Applications and Technologies* (2019).

[36] Xiaobo Xia, Tongliang Liu, Bo Han, Chen Gong, Nannan Wang, Zongyuan Ge, and Yi Chang. 2021. Robust early-learning: Hindering the memorization of noisy labels. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net. https://openreview.net/forum?id=Eql5b1_hTE4

[37] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. 2015. Learning from massive noisy labeled data for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. IEEE Computer Society, 2691–2699. https://doi.org/10.1109/CVPR.2015.7298885

[38] Jiangchao Yao, Jiajie Wang, Ivor W. Tsang, Ya Zhang, Jun Sun, Chengqi Zhang, and Rui Zhang. 2019. Deep Learning From Noisy Image Labels With Quality Embedding. *IEEE Trans. Image Process.* 28, 4 (2019), 1909–1922. https://doi.org/10.1109/TIP.2018.2877939

[39] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2017. Understanding deep learning requires rethinking generalization. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net. https://openreview.net/forum?id=Sy8gdB9xx

[40] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond Empirical Risk Minimization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net. https://openreview.net/forum?id=r1Ddp1-Rb

[41] Xiaodong Zhang and Houfeng Wang. 2016. A Joint Model of Intent Determination and Slot Filling for Spoken Language Understanding. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, Subbarao Kambhampati (Ed.). IJCAI/AAAI Press, 2993–2999. http://www.ijcai.org/Abstract/16/425

[42] Zhilu Zhang and Mert R. Sabuncu. 2018. Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (Eds.). 8792–8802. https://proceedings.neurips.cc/paper/2018/hash/f2925f97bc13ad2852a7a551802feea0-Abstract.html

[43] Guodong Zhou and Jian Su. 2002. Named Entity Recognition using an HMM-based Chunk Tagger. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*. ACL, 473–480. https://doi.org/10.3115/1073083.1073163

[44] Wenxuan Zhou and Muhao Chen. 2021. Learning from Noisy Labels for Entity-Centric Information Extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (Eds.). Association for Computational Linguistics, 5381–5392. https://doi.org/10.18653/v1/2021.emnlp-main.437