Customer Support Chat Intent Classification using Weak Supervision and Data Augmentation

Sumanth Prabhu[*]
Aditya Kiran Brahma[*]
sumanth.prabhu@swiggy.in
aditya.brahma@swiggy.in
Applied Research, Swiggy
Bangalore, Karnataka, India

Hemant Misra
hemant.misra@swiggy.in
Applied Research, Swiggy
Bangalore, Karnataka, India

ABSTRACT

Understanding the actual intent of customers is an essential step in automating the conversational experience on a chat platform. Typically, chatbots are powered by machine learning algorithms that rely on the acquisition of a large amount of high quality labeled training data which can be prohibitively expensive. To overcome this dependence on labeled training data, weaker forms of supervision have been recently exploited to generate samples in a more cost effective manner though the samples may be noisy. In this paper, we analyse a use-case specific to food delivery services where customer-agent conversations in incoherent English and code-mixed language (Hindi mixed with English, commonly referred to as Hinglish) are associated with a single customer chosen noisy label (referred to as "Conversation Level Intent" in the current paper) for the entire conversation. However, in reality, a conversation can have several messages and can be comprised of multiple labels. Moreover, each label may be associated with one or more messages in the conversation. In this paper, we demonstrate how simple light-weight word embeddings based weak supervision techniques can be used to tag individual customer messages with the most relevant label. We also show that simple augmentation techniques can significantly improve performance on code-mixed messages. On an internal benchmark dataset, we show that our sampling approach achieves an absolute performance gain of 33% in F1 score on random sampling strategy, 19% in F1 score over an approach using entire raw samples and 4% over a Snorkel (a state-of-the-art weak supervision framework) based approach.

KEYWORDS

Weak Supervision, Text Classification

ACM Reference Format:
Sumanth Prabhu, Aditya Kiran Brahma, and Hemant Misra. 2022. Customer Support Chat Intent Classification using Weak Supervision and Data Augmentation. In *5th Joint International Conference on Data Science & Management of Data (9th ACM IKDD CODS and 27th COMAD) (CODS-COMAD 2022), January 8–10, 2022, Bangalore, India.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3493700.3493729

1 INTRODUCTION

In this era of smart conversational agents for day to day services, Natural Language Understanding (NLU) techniques [8, 36] are extensively used to extract the semantic representation of a user's message. [18] explored the possibility of personalizing customer support experience for each customer. A major requirement for these supervised approaches to perform well is the availability of abundant labelled training data.

In this paper, we consider the use-case of customer support for food delivery services where customers can place orders across restaurants and reach out to customer support for issues faced by them related to their orders. Table 1 shows a sample conversation where a customer and a customer support agent exchange messages. Typically, a customer selects a particular issue they are facing for the chosen order from a pre-defined set of issues, and requests to chat with an agent. We refer to the issue chosen by the customer as the "Conversation Level Intent" (CLI). A pitfall of this methodology is that the entire conversation is associated with only one such CLI. In other words, this methodology assumes that the customer's intent remains the same throughout the conversation. In reality, the customer's intent may change during the conversation without any explicit signals, and thus, cannot be resolved using the CLI alone. To be able to automate the conversational experience, intent classification must then operate at the message level to track changes in a customer's intent. Labelling this conversation data manually can be expensive while still not guaranteeing clean labels.

In a country like India where several languages are spoken, understanding the customers' intent becomes even more challenging as messages are often incoherent with different inflectional and spelling variations due to the wide-ranging literacy and English language fluency levels. Moreover, the messages can be code-mixed [32], thus, requiring a deeper semantic understanding. We analyze this code-mixed setting where we have noisy CLIs and learn to predict the message level intents. We focus on the use-case

[*] Both authors contributed equally to this research.

CODS-COMAD 2022, January 8–10, 2022, Bangalore, India
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-8582-4/22/01…$15.00
https://doi.org/10.1145/3493700.3493729

Message	Sender	Conversation Level Intent (CLI)	Message Level Intent (MLI)
“Hi ! I see you’ve raised a concern towards the order status. I assure you that I will do my best to assist you with your concern”	Agent	my order has not been picked up by the delivery executive yet	
“order pick up nhi hua abhi tak. can you call someone and check what is happening ?” **Translation :** “order is not picked up yet. can you call someone and check ?”	Customer		my order has not been picked up by the delivery executive yet
“Sure. Let me check this for you. Please give me a minute, I will be right back. Kindly stay connected.”	Agent		
“Thank you for staying connected. I can see that your order has been picked up and should reach you by 430PM”	Agent		
“I am trying to call the delivery partnet. He is not picking up. Can you check ?”	Customer		i am unable to reach the delivery executive

Table 1: Sample Conversation with customer chosen “Conversation Level Intent ” and customer “Message Level Intent”. All Hindi words have been underlined and the same convention has been followed throughout this paper

of Hindi mixed with English. [1] In this paper, we employ weak supervision and data augmentation approaches to extract weakly labeled training data for supervised intent classification of code-mixed queries. We then run experiments on the weakly labelled samples using both traditional machine learning approaches like Logistic Regression as well as state-of-the-art deep learning approaches like RoBERTa [43] and XLM-RoBERTa (XLM-R) [5] to verify the efficacy of our approach.

The main contributions of this paper are:

- We propose an intent classifier that leverages weak supervision and simple data augmentation techniques to handle both incoherent English and code-mixed customer messages
- Our method uses customer chosen CLI and derives robust labels at the message level
- We show the generalization of the approach by reproducing it with a different set of CLIs available to our customers

All the experiments are on real chat data. The rest of the paper is organized as follows: In Section 2 we describe previous work for solving similar problems. Section 3 describes our weak supervision based sampling approach in detail followed by Section 4 where we explain the experiment settings to validate our approaches with the corresponding results. Finally, we present our findings and describe future work in Section 5.

2 RELATED WORK

Weak supervision [7] and semi supervision [16] based sampling gained a lot of popularity given the benefits over regular supervised approaches. [9, 13] explored topic modelling based approaches to predict labels for documents. This approach may not work in our problem setting where customer messages are typically short. [37, 38] learned word embeddings by constructing word networks. [38] focused on learning dimension aware embeddings to perform document classification while [37] combined the available labeled data with the unlabeled data to learn word embeddings. [35] leveraged unlabelled data to pre-train models and then finetuned on well-labelled data. In contrast, our approach does not rely on the availability of any labelled data. We focus on expanding a minimal list of seed words for each label and discover relevant training data.

Similarly, [14] leveraged a semi-supervised approach to labelling documents where seed words are used to initialize labels to a set of documents. Our work is based on weak supervision where the relevant samples are labeled using keyphrases extended from a limited list of seed phrases. WeSTClass [20] generated pseudo documents using seed information and then bootstraped on real unlabeled documents using self-training. [24] further improves self training by incorporating uncertainty estimates of the underlying neural network. In this paper, we propose to train a supervised classifier by discovering training samples using weak signals.

In a different context, [4, 21, 25] looked at building weakly supervised frameworks focused on theoretical guarantees. We focus on a practical industry application of weak supervision where a single label exists for a conversation which may not be relevant to one or more messages in the conversation. [34] considered solving such problems by correcting the labels using an active learning algorithm to identify the optimal denoising function. We propose a different approach and focus on identifying the relevant messages for a label instead of correcting the label. Snorkel [29] framework combined various supervision sources and resolved conflicts using a generative model to assign a single label to each training sample. Our approach leverages the customer chosen Conversation Level intent to resolve conflicting labels.

ConWea [19] improved upon the previous weak labelling approaches by using BERT [6] based contextual embeddings to generate pseudo labels for document classification. Their idea was to

[1] Though Hindi and English both fall under the Indo-European language group, English is much closer to other European languages such as German, Dutch, Danish, Swedish etc.,(all fall under Germanic language group) and Latinate and French from which it has borrowed extensively [15] as compared to Hindi. In this context, Hindi written in Roman script and mixed with English creates new challenges as compared to mixing of English with other European languages which have a common script, common ancestor and similar root words.

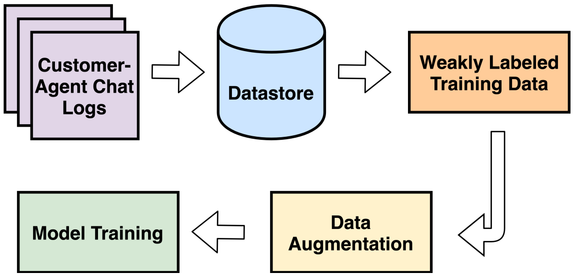

Figure 1: High level overview of the steps involved in data preparation for training an intent classification model

distinguish the representation of the candidate words per class based on the context of their usage. We show that pre-trained contextual embeddings do not work in our problem setting, thus, demanding customization. Our work generates weak labeled data by relying on external signals to disambiguate candidate keyphrases and seamlessly integrates with contextual embedding based supervised classification.

3 METHODOLOGY

In this section, we talk about the different approaches considered in this paper to extract weakly labeled data for training intent classification model. Figure 1 shows the high level overview of the system.

3.1 Weak supervision using keyphrases

Let I denote the list of possible intents that can be expressed by a customer. A customer chooses a particular CLI before starting the conversation with an agent. This intent maps to one of the intents from I. We observe that customer intents are often highly correlated with certain n-grams in English which customers typically use to express that intent. We create a minimal list of such n-grams for each intent using subject matter expertise derived from analyzing customer-agent conversations. In the rest of the paper, the n-grams and the list of n-grams are referred to as the "keyphrases" and the "seed keyphrase list" (denoted by S), respectively.

The seed keyphrase list is only a subset of the possible ways a customer can express the intent using free text during the chat conversation. To be able to capture multiple variations of the keyphrases present in S, we train a Word2Vec (W2V) model [23] using 171,237 messages from customer-agent conversations. We then generate representations for all words based on the contexts in which they have been used. For each keyphrase in the seed keyphrase list S, we identify synonyms with similarity scores greater than an empirically chosen threshold $\gamma_1 = 0.6$ and create a new list S'.

One of the major problems with chat conversations is that the customers tend to use multiple inflectional and spelling variations of keyphrases to express their intents. We hypothesized that FastText [3] would give more weightage to the inflectional and spelling variations because the training objective is based on generating embeddings for subword units [3] as opposed to W2V which captures embeddings at the word level [23]. For any given keyphrase, the top synonyms identified by W2V were paraphrases while the top synonyms identified by FastText were inflectional and spelling variations. Table 2 shows a few examples of keyphrases for the intent "the delivery executive is moving in the wrong direction". For each of the keyphrases in S', we identified FastText based synonyms with similarity scores greater than an empirically chosen threshold $\gamma_2 = 0.85$ and expanded the list of keyphrases to S''.

Pre-trained embeddings like BERT [6] have proven to be very successful with contextual weak supervision [19]. However, we observe that our problem setting demands customized models for weak supervision based sampling of training data. To validate our hypothesis, we ran a clustering experiment on a manually labelled set of 2,048 sample messages with 7 labels (CLI) and benchmark the performance of BERT embeddings vis-a-vis W2V embeddings. Concretely, we compute embeddings for each of the samples using different approaches and cluster the samples using the agglomerative clustering API provided by scikit-learn [26]. The number of clusters is set to 7 to match the number of labels in the test set. We use adjusted mutual information (AMI) [40] to measure the performance of the approaches. Figure 2 shows the clustering performance of pre-trained BERT based embeddings in comparison to that of the multiple weighing strategies using custom trained W2V embeddings. The weighing strategies explored in this paper were TF-IDF [33], Part-of-Speech (POS) tags and keyphrases (S''). With the POS tags strategy, we only consider the nouns and verbs to construct the sentence embeddings. We observe that pre-trained BERT embeddings perform poorly compared to custom W2V embeddings. With W2V, we observe that computing the sentence embeddings with the identified keyphrases gives the best performance. We also experimented with Sentence BERT (SBERT) [30] which has proven to achieve good performance on sentence representations. We experiment with XLM-R [5] embeddings trained on 100 different languages including romanized Hindi. However, both these embeddings also perform worse than W2V based embeddings for the given problem setting. Additionally, we observe that BERT based embeddings weighted with keyphrases perform better than W2V based embeddings. Hence, we leverage BERT based models for the downstream intent classification model training once the keyphrases have been identified.

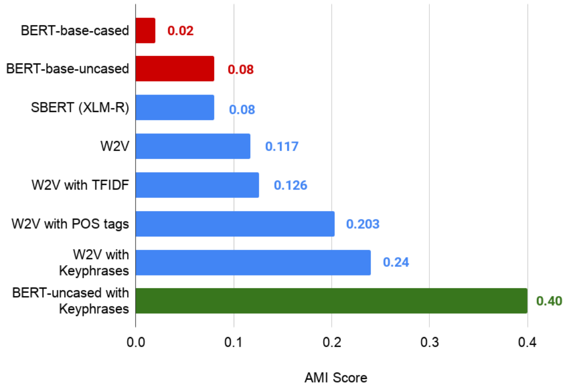

Figure 2: Bar plot of adjusted mutual information (AMI) scores for different clustering approaches

W2V	FastText
“different direction”	*“wrng direction”*
“wrong way”	*“worng direction”*
“circles”	*“wrond direction”*

Table 2: Sample synonyms from W2V and FastText for kephrase “wrong direction”

3.2 Conflict Resolution

Using the keyphrases in S'', we then search for customer messages in the conversations that contain these keyphrases and include such messages into the pool of weakly labeled training data. However, there can be cases where certain keyphrases match multiple intents. Table 3 shows one such example.

Message	Intent	Keyphrase
“order pick up nhi hua abhi tak. can you call someone and check ?” **Translation :** “order is not picked up yet. can you call someone and check ?”	my order has not been picked up by the delivery executive yet	picked up;call
“I am trying to call the delivery partnet. He is not picking up. Can you check ?”	i am unable to reach the delivery executive	call; picking up

Table 3: Sample customer messages with same key phrases for different intents

Our problem setting includes a weak signal coming from the customer in the form of the chosen CLI. We filter messages detecting presence of keyphrases from S'' that map to the chosen CLI. This not only strengthens our confidence in the message being relevant to the true customer intent but also helps resolve labelling of messages with conflicting intents. Messages with conflicting intents are cases where a message can get mapped to different intents due to the presence of overlapping set of keyphrases. For messages that have an intent matching the intent of the conversation, we assign that CLI as the intent for that message and discard the messages for which none of the intents match the CLI. For example, to construct the training dataset for intent “i am unable to reach the delivery executive”, if we encounter a message containing the keyphrase “picking up” and the chosen intent for the conversation is “i am unable to reach the delivery executive”, we include the message in the pool of training samples. If the keyphrase does not map to the intent or the CLI does not match the intent, we reject the message.

3.3 Augmentation using Lexical Substitution

Another major challenge with customer chat intent classification is the occurrence of code mixed messages. The keyphrase based weak supervision strategies described in the previous section work well for incoherent English messages but fail for code-mixed messages where the keyphrases are not expressed in English. The problem is more prononunced when the languages being mixed are not very similar, for example, Hindi mixed with English. In this paper, we consider only English-Hindi code mixing. W2V and FastText fail to capture Hindi keyphrases and hence such messages are not captured in the training samples identified.

To be able to incorporate Hindi synonyms for keyphrases identified in set S'', we employ an augmentation strategy using external Hindi synonyms (identified by internal language experts). We also explore additional English synonyms currently missing in the samples using WordNet [39] and further augment the keyphrases. Concretely, for every training sample identified, we replace the keyphrases used in the message with a newly identified external synonym for that particular intent and expand the training data.

4 EXPERIMENTS

In this section, we describe the dataset, the baselines and the evaluation metrics used to benchmark the experiments.

4.1 Dataset

The dataset considered for the experiment comprises of 513,484 conversations. For the purpose of our experiments, we consider the following 7 CLIs related to "Where is my order?":

- “Other”
- “i am unable to reach the delivery executive”
- “i am unable to track my delivery executive on order tracking page”
- “my order has not been confirmed by the restaurant”
- “my order has not been picked up by the delivery executive yet”
- “no delivery executive has been assigned to my order yet”
- “the delivery executive is moving in the wrong direction”

Customers have a tendency to use similar messages. After removing duplicate messages per customer chosen CLIs across conversations, the total number of unique messages available in the data sample is 308,343. We restrict to syntactic de-duplication and refrain from removing semantically duplicate messages to retain variants of customer behaviour in chat conversations.

Table 4 shows the number of training samples obtained by each of the approaches discussed in Section 3. Five people were chosen to label 1,648 customer messages sampled from the chat conversations to construct the test set for intent classification. In addition, 80 seed samples were randomly chosen from this test set to create 400 manually crafted paraphrases (pure English variations and Hindi-English variations) to increase the variations of the samples in the test set. The final test set is comprised of 2,048 samples. 31% of the test samples do not have any keyphrase identified by our approach in the train samples. In other words, our proposed approach scales to these unseen keyphrases.

4.2 Keyphrase construction

Table 5 shows the seed words created for the previously listed intents. For the purpose of the experiments in the paper, we restrict phrases to unigrams and bigrams. We manually identified 8 keyphrases in total. Table 6 shows the list of external synonyms created as described in Section 3.3. We identified 4 English synonyms and 4 Hindi synonyms in total. For intent “Other”, there are no specific keyphrases that can be mapped. We observe that the customer

Methodology	Train sample size	Test sample size
W2V	8,628	2,048
W2V + FastText	10,744	
Augmentation	24,087	
Snorkel	26,443	
Raw data	308,343	

Table 4: Dataset size for various sampling methodologies

Intent	Seed Words
my order has not been picked up by the delivery executive yet	picked up
the delivery executive is moving in the wrong direction	wrong direction
no delivery executive has been assigned to my order yet	assigned
my order has not been confirmed by the restaurant	confirmed
i am unable to reach the delivery executive	reach
	picked up
	call
i am unable to track my delivery executive on order tracking page	track

Table 5: Manually constructed list of Seed Keyphrases per Conversation Level Intent (CLI)

messages in a conversation rejected during the construction of the training data for the remaining intents have a high probability of belonging to an irrelevant intent. Hence, we consider the strategy of sampling a fraction of such rejected messages to match the distribution of the customer chosen CLIs across conversations and label them with the intent "Other".

Intent	English Synonyms	Hindi Synonyms
my order has not been picked up by the delivery executive yet	collected	uthaya,liya
the delivery executive is moving in the wrong direction	wrong way	galat direction
no delivery executive has been assigned to my order yet	designated, allotted	
i am unable to reach the delivery executive		uthaya

Table 6: Manually constructed list of external synonyms for "Where is my order?"

4.3 Baselines

Raw data The first baseline is a traditional supervised setting which labels all the messages in a conversation with customer chosen CLI. We assume that the CLI is the true intent of the customer and every customer message sent during the conversation maps to that intent.

Snorkel To validate the efficacy of our approach, we experimented with the Snorkel [29] framework where multiple correlated sources for data labelling are applied to obtain a single noisy label for each training sample using a generative model [1, 2]. Keyphrases from W2V and FastText are used to construct the labelling functions. A snorkel approach with additional regex patterns achieved on-par performance with the current approach. However, it involved a lot of manual effort in constructing the relevant regex patterns. Hence, we do not consider it as a baseline.

4.4 Experiment Setting

W2V and FastText are trained with $embedding_dim$ set to 100. γ_1 and γ_2 were set to 0.6 and 0.85 respectively. The thresholds were empirically set to control the keyword count and achieve high clustering performance on manually labelled samples. We retained the same thresholds for remaining intents and observed similar levels of accuracy. W2V models were trained with CBOW [22].

Datasets obtained for different sampling approaches were used to train the following three models - Logistic Regression, RoBERTa [43] and XLM-R [5]. Logistic Regression was chosen to evaluate the performance of the data sampling approaches from a traditional machine learning perspective. RoBERTa has been proven to achieve state-of-the-art results on GLUE [41], RACE [11] and SQuAD [27, 28]. XLM-R pre-trained on 100 different languages including romanized Hindi was included to understand the impact of pre-trained multilingual language models in code-mixed settings [5].

Logistic regression is combined with CountVectorizer [12] with parameter min_df set to 10 to extract features from the training samples. RoBERTa and XLM-R are run with $batch_size$ set to 64, $embedding_dim$ set to 768 and $epochs$ set to 5. $maxseq_length$ was set to 128. We use the Adam [10] optimizer with $epsilon$ set to 1e-8 and $learningrate$ set to 4e-5. For RoBERTa, we initialized the embeddings from roberta-base model provided by huggingface [42] which has 12 layers, 12 heads, 125M parameters and an embedding dimension of 768 . For XLM-R, we initialized the embeddings from xlm-roberta-base provided by huggingface [42] which has 125M parameters with 12 layers, 8 heads, embedding dimension of 768 trained on 2.5 TB of newly created clean CommonCrawl data in 100 languages..

4.5 Results and Discussion

Table 7 shows the F1-scores of the three classification models across the previously described sampling approaches.

Training models with the full conversations without any sampling of messages yields similar performance across all the models. Weak supervision with W2V based sampling appraoch performs better than using the entire raw data as samples. Even though the number of samples with this approach is less than 3% of the number of messages in the conversations, we see an increase in F1-scores because of retaining only those samples that are relevant to the CLI.

Weak supervision using both W2V and FastText based sampling approaches perform better than the W2V based sampling approach. We believe this is because of the additional variations of keyphrases captured by FastText described in Section 3.

Models based on training data extracted using Snorkel with keyphrase based labelling functions outperform models trained on raw data as samples across all models but underperforms relative to W2V and FastText based weak supervision sampling approaches. This shows that in the presence of multiple candidate labels, using the customer chosen CLI to resolve conflicts works better than generative model based approaches. Data augmentation with external Hindi and English synonyms for keyphrases outperforms all other strategies by accounting for code-mixed messages. Logistic regression performs worse with augmented data while RoBERTa and XLM-R show an improvement in performance. We believe that this is because RoBERTa and XLM-R are based on character language models which have been proven to account for inflectional and spelling variations over traditional approaches [31], thus, giving better performance.

Performance comparison of English vs code-mixed queries The test set considered for the experiments comprises of 88.62% queries in English and 11.38% code-mixed queries with keyphrases expressed in Hindi. Tables 8 and 9 show the performance of the models for English and code-mixed queries respectively. We observe that Logistic Regression performs really well with W2V + FastText based weak supervision. Augmentation with external English synonyms boosts the performance for RoBERTa by one point. Similarly, for code-mixed queries, we observe a significant performance increase for all three models with data augmentation with external English and Hindi synonyms.

Approach	**Models**		
	LR	**RoBERTa**	**XLM-R**
Raw data	0.74	0.72	0.71
Snorkel	0.83	0.87	0.87
W2V	0.80	0.86	0.87
W2V + FastText	0.89	0.89	0.89
Augmentation	0.85	**0.91**	**0.91**

Table 7: F1-scores (English and code-mixed combinned) across various sampling approaches and classification approaches

Approach	**Models**		
	LR	**RoBERTa**	**XLM-R**
Raw data	0.77	0.74	0.75
W2V	0.85	0.91	0.91
W2V + FastText	**0.93**	0.93	0.93
Snorkel	0.87	0.91	0.92
Augmentation	0.87	**0.94**	**0.93**

Table 8: F1-scores for English queries across various key phrase generated methodologies and models

Approach	**Models**		
	LR	**RoBERTa**	**XLM-R**
Raw data	0.5	0.51	0.57
W2V	0.31	0.46	0.42
W2V + FastText	0.51	0.57	0.56
Snorkel	0.58	0.54	0.55
Augmentation	**0.65**	**0.68**	**0.7**

Table 9: F1-scores for code-mixed queries across various key phrase generated methodologies and models for code-mixed queries

Message	**Label**	**Sampling Strategy**	
		A1	A2
"ladka app main nahi dikh ra" **Translation:** *"Guy not visible on the app"*	i am unable to track my delivery executive on order tracking page	×	✓
"abhi tak nahi uthhaya" **Translation:***"not yet picked up"*	my order has not been picked up by the delivery executive yet	×	✓

Table 10: Sample message predictions from XLM-R for "W2V+FastText" (A1) vs "Augmentation" (A2)

Sample message predictions Table 10 shows how augmentation helps handle code mixed samples better than just keyphrase based samples. Moreover, RoBERTa and XLM-R robustly handle new spelling variations of Hindi words not included during Augmentation. Table 11 shows examples of different scenarios where each model fails or succeeds to predict the right intent with augmented samples. RoBERTa and XLM-R learn from the augmented samples and perform well with test samples not covered by keyphrases based on W2V and FastText. Although the overall performance of both these models based on F1-scores is the same, we do observe cases where XLM-R handles spelling variations of keyphrases better than RoBERTa. Table 12 shows similar examples for each model with Snorkel based training data samples. We notice that RoBERTa and XLM-R do a really good job but for keyphrases like 'pick up' which are common to multiple intents ('order not picked up' vs. 'call not picked up'), XLM-R does a better job of understanding the context. We also notice samples where all models still fail in each of the approaches. These are edge cases and would require further analysis which we will be addressing in future work.

To understand the significance of our results, we leverage Stuart-Maxwell test [17] to compare RoBERTa's predictions using raw samples with the predictions using augmentation. We observe a chi-squared statistic of 329.84 with 6 degrees of freedom giving a p-value of 2.2e-16. In other words, the improvement achieved by

Message	Label	Model		
		M1	M2	M3
“Restaurant not even confirmed the order till now”	my order has not been confirmed by the restaurant	✓	✓	✓
“haan order collect nahi hua ab tak” **Translation:** *"yes order not collected yet"*	my order has not been picked up by the delivery executive yet	×	✓	✓
“Order nahi accpet hua” **Translation:** *"Order not accepted"*	my order has not been confirmed by the restaurant	×	✓	✓
“order tayar hai pur koi lene hi nahi pahucha - chul kya raha hai?” **Translation:** *"order is ready but noone going to pick up - what is going on ?"*	my order has not been picked up by the delivery executive yet	×	×	×

Table 11: Sample message predictions for LR (M1), RoBERTa (M2) and XLM-R (M3) with sampling approach "Augmentation"

performing sampling with weak supervision and data augmentation approach is significant with respect to the baseline.

Random sampling strategy To compare the current sampling strategy and the quality of data generated by it, we considered random sampling strategy where the number of samples are equal to the number of samples generated from the proposed approach in the paper. Logistic regression models were built on 100 different datasets which were randomly sampled. We calculated the performance of these models on the available test data and the average of the F1 scores from the 100 models is 0.57. Due to experimental resource constraints, we trained XLM-R only on one of the randomly sampled datasets, it achieved an F1 score of 0.58. We observe that the performance of these models is significantly lower than the performance of the models built with sampling strategy proposed in the paper as well as the raw data.

To further understand the strengths and weaknesses of the approaches, we analyze the model performances using intent wise F1-score for raw data and data augmentation based approaches. Table 13 shows F1-scores per intent under “Where is my order?”. We observe that the major drawback with raw samples is that model fails to learn to identify the intent “Other”. Even if this intent is excluded, the augmentation approach still seems to do a better job due to the reduction in the number of noisy training samples.

Message	Label	Model		
		M1	M2	M3
“How shall i confirm whether my order has been accepted by the restaurant?”	my order has not been confirmed by the restaurant	✓	✓	✓
“delivery guy ko contact nhi kr pa rha hoon” **Translation:** *"not able to contact the delivery guy"*	i am unable to reach the delivery executive	×	✓	✓
“pick up location se bahut dur hai delivery executive” **Translation:** *“delivery executive is very far from the pickup location”*	my order has not been picked up by the delivery executive yet	×	×	✓
“das minte hogaya! Order bana kya? collect hua?” **Translation:** *"10 minutes have passed ! is the order prepared ? collected ?"*	my order has not been picked up by the delivery executive yet	×	×	×

Table 12: Sample message predictions for LR (M1), RoBERTa (M2) and XLM-R (M3) using Snorkel

4.6 Extending to other customer intents

To validate the reproducibility of the weak supervision and augmentation approach, we apply a similar strategy to intents related to “I have received bad quality food”. Following are the 9 CLIs considered for the experiment -

(a) *“My food was cold*, (b) *“I found unwanted ingredients in my food*, (c) *“My food was stale*, (d) *“My food was uncooked*, (e) *“My food was burnt”*, (f) *“My food was oily”*, (g) *“My food was bland”*, (h) *“My food was spicy”*, (i) *“My food was melted”*.

Table 14 shows a few example messages from customers describing bad quality food.

In addition to the challenges already described, “I have received bad quality food” is a multi-label problem. In other words, customers have a higher tendency to have more than one intent for a given message as opposed to “Where is my order?”. However customers are still restricted to choose only one intent for the entire conversation. Hence, we apply a similar strategy of sampling messages using keyword construction for each of the above intents with the key difference that we do not use the CLI to help resolve conflicts as described in Section 3.2.

We construct a training data set with 23,748 samples and a validation set with 2,640 samples with a setting similar to messages under “Where is my order?”. Table 15 shows the F1-scores per intent using our sampling approach with the RoBERTa model. We observe that the approach described in Section 3 is reproducible and extensible to more customer intents.

Intent	Sampling Approach	
	Raw Data	Augmentation
Other	0.15	0.85
i am unable to reach the delivery executive	0.82	0.88
i am unable to track my delivery executive on order tracking page	0.89	0.93
my order has not been confirmed by the restaurant	0.86	0.93
my order has not been picked up by the delivery executive yet	0.69	0.92
no delivery executive has been assigned to my order yet	0.85	0.95
the delivery executive is moving in the wrong direction	0.73	0.93

Table 13: F1-scores per intent for "Where is my order" related intents using "Raw data" and "Augmentation" sampling approaches with "XLM-R"

Message	Intent
"maine chicken order kia or chicken jala hua or oily aaya" **Translation :** I ordered chicken and the chicken received was burnt and oily]	My food was burnt; My food was oily
"i have found insect in the pizza"	I found unwanted ingredients in my food

Table 14: Sample customer messages for intents related to "I have received bad quality food"

Intent	F1-score
My food was cold	0.80
I found unwanted ingredients in my food	0.98
My food was stale	0.91
My food was uncooked	0.89
My food was burnt	0.92
My food was oily	0.77
My food was bland	0.82
My food was spicy	0.90
My food was melted	0.82

Table 15: F1-scores per intent for "I have received bad quality food"

5 CONCLUSION

In this paper, we introduce multiple weakly supervised approaches to identify relevant training samples for intent classification. We show that such approaches can reduce training costs and time while also increasing the accuracy of the model predictions. We also show how simple lexical substitution based data augmentation can increase the performance of the state-of-the-art modeling approaches especially for code-mixed messages involving low resource languages which do not share common roots with English. Overall, we show that our sampling approach achieves an absolute performance gain of 33% in F1 score on random sampling strategy, 19% in F1 score over an approach using entire raw samples and 4% over Snorkel. We also observed a performance gain of 13% when checked on code mixed queries explicitly. Moreover, all the weak supervision techniques presented are applicable to any text classification problem. We also observed that pre-trained BERT based models failed in data sampling. In future work, we plan to experiment with custom trained BERT models for sampling training data.

REFERENCES

[1] Stephen H. Bach, Daniel Rodriguez, Yintao Liu, Chong Luo, Haidong Shao, Cassandra Xia, Souvik Sen, Alexander Ratner, Braden Hancock, Houman Alborzi, Rahul Kuchhal, Christopher Ré, and Rob Malkin. 2019. Snorkel DryBell: A Case Study in Deploying Weak Supervision at Industrial Scale. In *ACM Special Interest Group on Management of Data.*

[2] Sonia Badene, Kate Thompson, Jean-Pierre Lorré, and Nicholas Asher. 2019. Data Programming for Learning Discourse Structure. In *Association for Computational Linguistics.*

[3] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching Word Vectors with Subword Information. In *Transactions of the Association for Computational Linguistics.*

[4] Nontawat Charoenphakdee1, Jongyeong Lee1, Yiping Jin, Dittaya Wanvarie, and Masashi Sugiyama. 2019. Learning Only from Relevant Keywords and Unlabeled Documents. In *Empirical Methods in Natural Language Processing.*

[5] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised Cross-lingual Representation Learning at Scale. In *Association for Computational Linguistics.*

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.*

[7] Kuzman Ganchev, João Graça, Jennifer Gillenwater, and Ben Taskar. 2010. Posterior Regularization for Structured Latent Variable Models. *J. Mach. Learn. Res.* 11 (Aug. 2010), 2001–2049.

[8] Sonal Gupta, Rushin Shah, Mrinal Mohit, Anuj Kumar, and Mike Lewis. 2018. Semantic Parsing for Task Oriented Dialog using Hierarchical Representations. In *Conference on Empirical Methods in Natural Language Processing.*

[9] Swapnil Hingmire and Sutanu Chakraborti. 2014. Topic labeled text classification: A weakly supervised approach. In *ACM SIGIR Conference on Research and Development in Information Retrieval.*

[10] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations.*

[11] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding Comprehension Dataset From Examinations. *arXiv preprint arXiv:1704.04683* (2017).

[12] Gilles Louppe Lars Buitinck, Fabian Pedregosa Mathieu Blondel, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. 2013. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning.* 108–122.

[13] Chenliang Li, Jian Xing, Aixin Sun, and Zongyang Ma. 2016. Effective document labeling with very few seed words: a topic modeling approach. In *ACM International on Conference on Information and Knowledge Management.*

[14] Ximing Li and Bo Yang. 2018. A Pseudo Label based Dataless Naive Bayes Algorithm for Text. In *International Conference on Computational Linguistics.*

[15] Ian MacKenzie. 2012. English as a lingua franca in Europe: bilingualism and multicompetence. *International Journal of Multilingualism* 9, 1 (2012), 83–100. https://doi.org/10.1080/14790718.2011.610506

[16] Gideon S. Mann and Andrew McCallum. 2007. Simple, Robust, Scalable Semi-Supervised Learning via Expectation Regularization. In *Proceedings of the 24th International Conference on Machine Learning* (Corvalis, Oregon, USA) *(ICML '07)*. Association for Computing Machinery, New York, NY, USA, 593–600. https://doi.org/10.1145/1273496.1273571

[17] A. E. Maxwell. 1970. Comparing the Classification of Subjects by Two Independent Judges. *British Journal of Psychiatry* 116, 535 (1970), 651–655. https://doi.org/10.1192/bjp.116.535.651

[18] Pierre-Emmanuel Mazaré, Samuel Humeau, Martin Raison, and Antoine Bordes. 2018. Training Millions of Personalized Dialogue Agents. In *Conference on Empirical Methods in Natural Language Processing*.

[19] Dheeraj Mekala and Jingbo Shang. 2020. Contextualized Weak Supervision for Text Classification. In *Association for Computational Linguistics*.

[20] Yu Meng, Jiaming Shen, Chao Zhang, and Jiawei Han. 2018. Weakly-Supervised Neural Text Classification. In *ACM International Conference on Information and Knowledge Management*.

[21] Aditya Menon, Brendan Van Rooyen, Cheng Soon Ong, and Bob Williamson. 2015. Learning from Corrupted Binary Labels via Class-Probability Estimation. In *Proceedings of Machine Learning Research*.

[22] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. arXiv:1301.3781 [cs.CL]

[23] Tomas Mikolov, Ilya Sutskever, Kai Chen, Dittaya Wanvarie, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems*.

[24] Subhabrata Mukherjee and Ahmed Hassan Awadallah. 2020. Uncertainty-aware Self-training for Text Classification with Few Labels. arXiv:2006.15315 [cs.CL]

[25] Nagarajan Natarajan, Inderjit S. Dhillon, Pradeep Ravikumar, and Ambuj Tewari. 2013. Learning with Noisy Labels. In *Neural Information Processing Systems*.

[26] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[27] Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Association for Computational Linguistics*.

[28] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Empirical Methods in Natural Language Processing*.

[29] Alexander Ratner, Stephen H. Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. 2017. Snorkel: Rapid Training Data Creation with Weak Supervision. In *Proceedings of the VLDB Endowment*.

[30] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (2019). https://doi.org/10.18653/v1/d19-1410

[31] Brian Riordan, Michael Flor, and Robert Pugh. 2019. How to account for misspellings: Quantifying the benefit of character representations in neural content scoring models. In *Proceedings of the 14th Workshop on Innovative Use of NLP for Building Educational Applications (BEA)*.

[32] Koustav Rudra, Shruti Rijhwani, Rafiya Begum, Kalika Bali, Monojit Choudhury, and Niloy Ganguly. 2016. "Understanding Language Preference for Expression of Opinion and Sentiment: What do Hindi-English Speakers do on Twitter?". In *Conference on Empirical Methods in Natural Language Processing*.

[33] Gerard Salton and Michael J McGill. 1986. Introduction to modern information retrieval. (1986).

[34] Karan Samel and Xu Miao. 2018. Active Deep Learning to Tune Down the Noise in Labels. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining)*.

[35] Xiaoming Shi, Haifeng Hu, Wanxiang Che, Zhongqian Sun, Ting Liu, and Junzhou Huang. 2020. Understanding Medical Conversations with Scattered Keyword Attention and Weak Supervision from Responses. *Proceedings of the AAAI Conference on Artificial Intelligence* 34, 05 (Apr. 2020), 8838–8845. https://doi.org/10.1609/aaai.v34i05.6412

[36] Aditya Siddhant, Anuj Goyal, and Angeliki Metallinou. 2019. Unsupervised Transfer Learning for Spoken Language Understanding in Intelligent Agents. In *Association for the Advancement of Artificial Intelligence*.

[37] Jian Tang, Meng Qu, and Qiaozhu Mei. 2015. PTE: Predictive text embedding through large-scale heterogeneous text networks. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[38] Fangbo Tao, Chao Zhang, Xiusi Chen, Meng Jiang, Tim Hanratty, Lance Kaplan, and Jiawei Han. 2018. Doc2cube: Automated document allocation to text cube via dimension-aware joint embedding. In *International Conference on Data Mining*.

[39] Princeton University. 2010. *"About WordNet"*. https://wordnet.princeton.edu/

[40] Nguyen Xuan Vinh, Julien Epps, and James Bailey. 2010. Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance. In *Journal of Machine Learning Research*.

[41] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *Empirical Methods in Natural Language Processing*.

[42] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. HuggingFace's Transformers: State-of-the-art Natural Language Processing. (2020). arXiv:1910.03771 [cs.CL]

[43] Myle Ott Yinhan Liu, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *Computing Research Repository* arXiv:1907.11692 (2019). https://arxiv.org/pdf/1907.11692.pdf version 1.