Main Predicate and Their Arguments as Explanation Signals For Intent Classification

Sameer Pimparkhede, Pushpak Bhattacharyya
Indian Institute of Technology Bombay
{sameerp, pb}@cse.iitb.ac.in

Abstract

Intent classification is crucial for conversational agents (chatbots), and deep learning models perform well in this area. However, little research has been done on the explainability of intent classification due to the absence of suitable benchmark data. Human annotation of explanation signals in text samples is time-consuming and costly. However, from inspection of data on intent classification, we see that, more often than not, the main verb denotes the action, and the direct object indicates the domain of conversation, serving as explanation signals for intent. This observation enables us to hypothesize that **the main predicate in the text utterances along with the arguments of the main predicate can serve as explanation signals**. Leveraging this, we introduce a new technique to automatically augment text samples from intent classification datasets with word-level explanations. We mark main predicates (primarily verbs) and their arguments (dependency relations) as explanation signals in benchmark intent classification datasets ATIS and SNIPS, creating a unique 21k-instance dataset for explainability. Further, we experiment with deep learning and language models. We observe that models that work well for classification do not perform well in explainability metrics like *plausibility* and *faithfulness*. We also observe that guiding models to focus on explanation signals from our dataset during training improves the *plausibility* Token F1 score by 3-4%, improving the model's reasoning.

1 Introduction

Intent classification is widely used in real-life chatbots like Alexa, Siri, and other conversational AI tools. It is used to grasp the nuanced aspects of user expressions. It involves (i) comprehending the intentions behind the expressions to guide the agent's actions and (ii) identifying slots that indicate crucial entities necessary for executing the actions. Intent identification is studied as a multiclass classification problem by Raymond and Riccardi (2007). However, it is observed that jointly solving it with slot filling improves the performance (Qin et al., 2021; Chen et al., 2019). While deep learning models perform well in intent classification for specific datasets (Chen et al., 2019; Gunaratna et al., 2022), they face challenges in generalizing across diverse domains and fine-grained intents, particularly for classes with scarce data (Elazar et al., 2021; Casanueva et al., 2020). In scenarios with numerous conversation domains and fine-grained intent classes, acquiring significant data for every class becomes challenging. Additionally, the presence of shared words among fine-grained classes with subtle variations in meaning adds complexity to the task. In the text *Find me an airline with a meal facility*, the intent is *atis-airline* as the focus is on the airline, not the meal. While *meal* is a key entity, it doesn't define intent. Conversely, in *What are my meal options on airlines from Boston to Denver*, the intent is *atis-meal* as the user asks about meals, not the airline. However, the deep learning model learns from correlations and frequently used words. Hence, it misclassifies the later utterance as atis-airline due to the abundance of that class in the training data and the higher frequency of the term *airline*. This leads to a low overall F1 score and low accuracy for classes with fewer samples (Table 4). These misclassifications severely affect task-oriented dialogue systems. For instance, the text *At the Charlotte airport, how many different types of aircraft are there for us?* is often misclassified by BERT as atis-airport instead of atis-aircraft due to an imbalance in sample sizes. This can mislead the conversation ahead. However, performance is improved if the model focuses on the right keywords, like aircraft. Due to this, it is essential to analyze the models' reasoning for intent classification and improve their performance.

Text	Where can I <mark>watch</mark> the <mark>television</mark> <mark>show</mark> The Private Affairs of Bel Ami ?
Label	SearchCreativeWork
Text	<mark>Book</mark> a <mark>table</mark> at the top-rated <mark>pub</mark> in Garner
Label	BookRestaurant
Text	What are my <mark>meal</mark> <mark>options</mark> on airlines from Boston 066 to Denver
Label	atis-meal
Text	what <mark>price</mark> is a <mark>limousine</mark> <mark>service</mark> to New York's La Guardian
Label	atis-ground-fare

Table 1: Examples of text utterances and ground truth Labels from SNIPS and ATIS datasets. The highlighted part denotes the explanation signals marked using the Main predicate and its arguments.

Unfortunately, even if we get the model's explanation for intent identification, there are no ground truth datasets to evaluate it. Generally, such datasets are created using human annotation, but it is costly and time-consuming (Mathew et al., 2021; DeYoung et al., 2019; Hayati et al., 2021). Recent methods use large language models (LLM) to generate synthetic explanations. This approach might provide unreliable explanations and, hence, is difficult to integrate into business (Li et al., 2022; Ye and Durrett, 2022). Hence, we create word-level explanation signals automatically using an interesting linguistic observation. We observe that the main verb denotes action, and its direct object denotes the domain of conversation. Hence, they can act as explanation signals for the intent class. Inspired by these observations, we introduce a novel silver standard annotation technique that uses dependency relations among words as shown in figure 1 and augments text samples from intent classification datasets with word-level explanation signals. Using this technique, we derive high-quality silver standard explanations for two intent classification datasets ATIS (Hemphill et al., 1990) and SNIPS (Coucke et al., 2018). Hence, we augment 21k text samples with explanation signals and create the first benchmark dataset for explainability in intent classification. We perform the human evaluation of silver annotations to verify the quality of automatically generated explanation signals. For examples from our dataset, refer to table 1.

Further, we investigate the reasoning of popular deep learning and language models. Explanations are derived from models in the form of saliency maps using standard post-hoc explanation techniques LIME (Ribeiro et al., 2016) and integrated gradients (Sundararajan et al., 2017). We observe that the models that work well for intent classification often do not perform well on explainability metrics *plausability* and *faithfulness* (refer to table 2 and 3). To improve the reasoning, we guide the model to focus more on silver annotated explanation signals from our dataset during training. It helps the model make more human-like decisions and improves performance for explainability metrics. Our system is ideal for chatbots in small enterprises that can't afford in-house and costly models like GPT-4. Instead, improving reasoning for smaller models like LSTM, BERT, and GPT2, which give efficient responses, can benefit intent classification in chatbots. One can quickly annotate the explanation signals using the main predicate and arguments in the ongoing conversation and guide the deep learning models to focus on those signals more.

Our contributions are:

1. A Novel technique to create silver annotations of word-level explanation signals using the main predicate and its arguments for the text samples from standard intent classification datasets (Figure 1 and section 4.2) with a focus on explainability, which is highly relevant for dialogue agents and chatbots used in small-scale enterprises.

2. A First-of-its-kind dataset with high-quality word-level explanations of the existing intent classification datasets ATIS and SNIPS consisting of $20k$ samples.

3. Detailed experiments to investigate the explainability performance in *plausibility* and *faithfulness* of deep learning and language models on our dataset (Table 2 and 3). We create the first benchmark for explainability in the domain of intent classification.
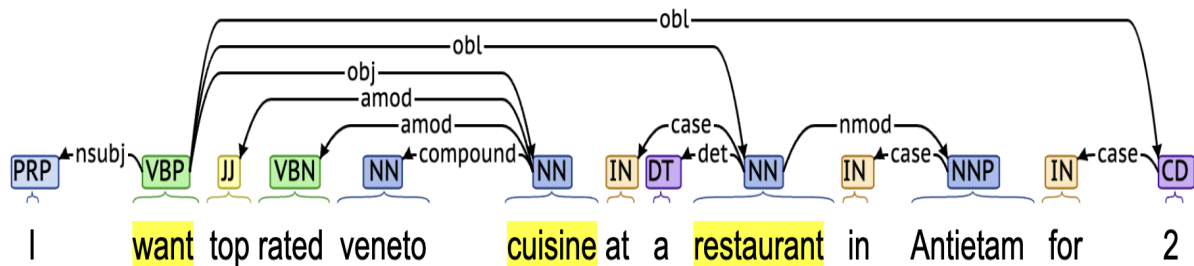
Figure 1: Dependancy parse tree of text sample from SNIPS dataset having a label as BookRestaurants. Highlighted tokens are marked as explanation signals according to the algorithm. This parse tree is fetched using Stanford CoreNLP with OpenIE in the background
.

2 Related work

Intent classification is a well-studied problem in natural language understanding (NLU). While intent classification is a sentence-level classification task, slot filling is a more challenging task that deals with classifying the type of each word. These problems can be solved independently Raymond and Riccardi (2007), but they are generally solved jointly to optimize performance Chen et al. (2019); Qin et al. (2021). Our primary focus here is to investigate and improve the model's reasoning for intent classification.

In explainability, post-hoc explanation techniques are popular and useful for feature attribution. This includes perturbation-based methods (Ribeiro et al., 2016, 2018), Gradient-based methods like integrated gradient (Sundararajan et al., 2017) and attention-based methods (Wu and Ong, 2021; Wang et al., 2016; Xu et al., 2015). For natural language processing (NLP) applications, feature importance is measured by the attribution score assigned to every token. These methods explainform of saliency maps, which is a suitable form for NLP due to well-defined features like words and phrases.

Many methods go beyond evaluation and ingest feature attribution priors during training. Some methods use attribution scores derived from post-hoc explanation techniques to train a hate-speech classifier under a scarce data scenario like Liu and Avci (2019). Zhong et al. (2019) and Mathew et al. (2021) supervised the model's attention using human-annotated rationale. Jayaram and Allaway (2021) incorporates feature attribution for documents from the legal domain. However, these methods primarily focus on tasks like hate speech or sentiment classification. We instead explore the

area of intent classification.

Regarding explainability in intent classification, Joshi et al. (2021) uses Layer wise relevance propagation (LRP) (Montavon et al., 2019) to investigate the deep learning model's reasoning over the ATIS dataset. However, this study only evaluates qualitatively based on some examples due to a lack of ground truth explanation signal data. To solve this problem, we introduce a benchmark dataset using the novel silver annotation technique and evaluate the models' reasoning quantitatively based on multiple metrics. Gunaratna et al. (2022) focuses on explaining slot classification, not intent.

3 Dataset

We use two intent classification benchmark datasets, ATIS (Hemphill et al., 1990), and SNIPS (Coucke et al., 2018). ATIS consists of a set of text utterances and their corresponding intent labels. It has 4478, 500, and 893 text utterances with gold intent in train, development, and test set, respectively. It consists of 21 intents about the airline travel domain. The second dataset, SNIPS (Coucke et al., 2018), consists of 13,084 training utterances; the validation and test sets consist of 700 utterances each, with 100 queries per intent. It has seven different intent classes, each belonging to a different domain. Text utterances are from domains like restaurants, movies, music, etc.

We select these datasets with a specific purpose. ATIS has fine-grained intents, and some classes have a low training and test set size (refer to table 4). This imbalance enables us to evaluate the generalizability and reasoning of the deep learning models when dealing with data scarcity. On the other hand, the SNIPS dataset is balanced but

consists of utterances from multiple domains. We can use the SNIPS dataset to evaluate the model's reasoning for intents from diverse domains. Evaluating the reasoning of deep-learning-based models on both aspects gives a good analysis.

We mark word-level explanation signals in both the ATIS and SNIPS datasets. Hence, we contribute around 21k samples with explanation signals. From ATIS, we filter out the utterances with more than one intent, and we also filter classes in the test set but not in the train set. The average length of silver annotated explanation for the SNIPS dataset is 3.4 words, while it is 4.2 for the ATIS dataset.

4 Silver annotated explanations

Human annotation of word-level explanations for intent classification is difficult. Deciding whether original slots in the dataset should be added as an explanation signal is subjective and generally requires multiple human annotators. For example, text utterance, *Show me the flights from Pittsburgh to Los Angeles on Thursday* has gold intent of *atis-flight*. In this example, Pittsburgh and Los Angeles are slots with slot-types source city and destination city. These entities are essential to execute appropriate further actions. However, it does not explain the intent because changing the city names to other cities does not affect the intent of the utterance. Generally, multiple humans annotate explanation signals, which is costly and time-consuming. Hence, we propose a method to introduce word-level explanation signals automatically to existing intent classification datasets consisting of text utterance and gold intent. We define our dataset as follows. N is the number of samples in the dataset, and each text sample X_i has m_i tokens denoted by $(w_1, w_2, w_3, ...w_{m_i})$. We add word-level explanation signals e denoted by $(e_1, e_2, e_3, ...e_{m_i})$ where each $e_j \in \{0, 1\}$. y_i is a gold intent label for every X_i, directly adopted from the original intent classification datasets. We add these word-level explanation signals for both ATIS and SNIPS datasets.

4.1 Using slots in dataset

ATIS and SNIPS datasets consist of slots and their type, which provide specific conversation-related information. Example: Text utterance "which airlines fly from Boston to Washington DC via other cities" has gold intent label as *ATIS-AIRLINE* and

slots marked in B-I-O format as "O O O O B-fromloc. city-name O B-toloc. city-name B-toloc. state-code O O O". Hence "Boston", "Washington" and "DC" are the words which are marked as slots. Although these slots provide essential entities to execute the actions, they fail to deliver an *explanation* for the gold intent atis-airline. Slots do not cover words like "airlines." These words are essential to provide an explanation as the dataset has other intent classes like *atis-aircraft* and *atis-flight*, which have very close meanings. Similarly, in the SNIPS dataset, text utterance *Add artist to playlist Epic Gaming has slots marked as "O B-music-item O O B-playlist I-playlist.* The slots in this text sample are *artist epic gaming*, which does not form a sufficient explanation for gold intent *AddPlaylist*. Moreover, intent does not change, even if we replace "epic gaming" with other playlist names. These examples depict that slots in the dataset can't be used as a sufficient and comprehensive explanation.

4.2 Main Verbs and Arguments

Creating silver standard explanation signals for intent classification comes from a simple yet significant insight. In many cases, information for classifying the intent of the text utterance is encoded within the main verb and its direct object. *Example: Set up an alarm for 9 AM tomorrow* has the intent *set alarm* and *delete all the alarms* has the intent *delete-alarm*. Here, the word *alarm* informs that action has to be performed on the alarm, and the words *set* and *delete*, which are main verbs, inform about a specific action to be performed. Now, if we change the domain from setting an alarm to setting a calendar, it is the direct object of the main verb that denotes this change in the domain of conversation. *Example: Set up a meeting for 9 AM tomorrow* has the intent *set calendar*. However, due to the diverse structures of human language, explanation signals might play different syntactic roles, and relying upon just the main verb and direct object may not provide sufficient explanation. Hence, we design a technique that considers multiple linguistic features using the dependency parse tree, automatically deriving high-quality explanations. The algorithm used to derive word-level explanation signals is as follows:

Algorithm

1. Find the dependency tree of a given text utterance using the OpenIE platform.

2. Find the main predicate.

3. Travel the dependency parse tree from the root.

4. Find children of the main predicate with relation direct object, noun- subject, xcomp (open clausal complement), obl (oblique nominal)

5. In case of multiple obl (oblique nominal) for root word, include the obl with part of speech as a common noun.

6. mark these nodes as explanation signals and level 1 traversal ends here.

7. Travel the children of nodes collected in level 1. If the child node has a relation as "compound" and its part of speech is not a proper noun, mark it as an explanation signal.

8. Include all the visited nodes in the explanation

9. Remove stopwords from the explanation segment.

The algorithm starts by default by selecting the main predicate as the explanation signal. The main verb is not mentioned in the text utterances like *Ground Transportation at Baltimore airport*. Hence, We use the Main Predicate instead of the main verb. It is not always necessary that the main verb or main predicate is enough to explain the action. Hence, we also decided to add the xcomp of the main predicate. xcomp acts as a secondary predicate when the main predicate does not add much value as an explanation signal. For example, the text utterance from the ATIS dataset, "I need to fly from Atlanta to Denver" has the main predicate predicted by the OpenIE platform as "need". But, it does not add much value as an explanation of intent. Instead, including the word "fly" which is xcomp of the main predicate, confirms the domain of conversation. We choose to eliminate all the adjective modifiers, nominal modifiers, and oblique nominal with part of speech as proper nouns. These words may provide some relevant information about the conversation but might not be appropriate as an explanation. For example, the text utterance *Please list all cheapest flights*

on United from Denver to Baltimore has gold intent *atis-flight*. The word *cheapest* is an adjective that might be essential to provide specific information. Still, even if we change the word "cheapest" to "costliest," it does not change the intent.

Even though the direct object of the main predicate plays an important role, it does not necessarily provide all the information about the domain of conversation. Instead, providing only that explanation signal might be misleading when two intents are from different domains with similar actions. For example, "Add this item to the grocery" suggests the user is talking about a grocery item, and another statement, "Add this item to a playlist," indicates the user is talking about music. Both the words grocery and playlist have an obl dependence on the main predicate here, which must be included to provide a sufficient and comprehensive explanation. Finally, we consider cases where the main predicate might come in between, and the subject with part of speech nouns is essential. After including all the explanation signals, we filter the stopwords to reduce noise further. We use Python NLTK library as a source of stopwords [1]. Table 1 shows some examples of added explanation signals. Figure 1 shows an example of producing explanation signals using a dependency parse tree.

4.3 Dataset Quality Evaluation

We evaluate the quality of silver explanation signals generated in the form of the main predicate and its arguments. It is a time-consuming process to evaluate all 21k samples. Hence, we select an equal number of samples for every class and perform the human evaluation for this subset. We provided human evaluators with only the silver annotated explanation segment instead of the complete text. We asked them to select an appropriate intent class, for example, instead of a full-text sample *Where can I watch the television show The Private Affairs of Bel Ami ?*. We provide evaluators only the silver annotated explanation *watch television show* and ask them to select the most appropriate intent label. Evaluators were provided brief definitions of intent classes along with some examples. Evaluation is done separately for ATIS and SNIPS datasets. We randomly selected 200 samples from both datasets. In the selection process, we ensure no class imbalance in

---

[1] https://www.nltk.org/

any datasets. **Each sample is evaluated by three evaluators.** All the evaluators are experts in linguistics and language. Two annotators are Ph.D. candidates in linguistics; the third has completed a master's in English. We use three evaluators per sample due to the numerous classes in both datasets. Particularly, ATIS has 21 fine-grained intent classes. Given subtle differences in meanings among some classes from ATIS, relying on one or two annotators may lead to biased or inconsistent evaluation scores. To mitigate subjectivity, three evaluators annotate each sample's label. Inter Annotator Agreement (IAA) was calculated using Fleiss' Kappa score. **IAA kappa score obtained is 0.67 for the ATIS dataset and 0.74 for the SNIPS dataset sample.** Fleiss' Kappa score between 0.61 and 0.8 is considered to be *substantial* agreement (Landis and Koch, 1977). We evaluate human evaluation accuracy against the gold labels from original datasets. **The average human accuracy obtained is 96.3% for the ATIS dataset and 97.8% for the SNIPS dataset.** All the annotators were paid as per industry standards and the suitability of the task.

Although most of the text samples are identified correctly by evaluators, we perform a detailed analysis of the human evaluation. We observe that text utterances marked incorrectly resemble other classes in terms of their ground truth meaning. For example, text utterance *show me all the Lufthansa flights between Philadelphia and Denver* has ground truth intent as *atis-flight* and derived explanation using tree traversal algorithm is *show Lufthansa flights*. Two out of three evaluators misclassified it as *atis-airline*. Because *Lufthansa* is the name of the airline company, it is difficult to identify the exact class among *atis-airline* and *atis-flight*. Due to minimal distinction in meanings, the most confused pairs of classes in ATIS were *(aits-airline, atis-flight)* and *(atis-flight, atis-flight-time)*. For the SNIPS dataset, *SearchCreativeWork* consists of text utterances related to movies and songs that resemble very closely to classes *SearchScreeningEvent* and *PlayMusic*. For example, text utterance *Find me the song called The Budapest Beacon* has ground truth intent in SNIPS as *SearchCreativeWork* and the derived explanation is *Find song*. All three evaluators marked intent as *PlayMusic* when provided with an explanation part only. For SNIPS, evaluators were most confused with *SearchCreativeWork*

and *SearchScreeningEvent* classes. For these text samples, annotating correct intent may not be possible even with full-text utterances. Hence, this depicts that explanations derived from the main verb and its arguments are indeed high quality.

5 Experiments

5.1 Incorporating feature attribution

It is observed that ingesting feature attributions during training helps to improve the model's reasoning. We investigate if the model's reasoning improves by guiding it to focus on the silver annotated explanation signals. Following the work of Liu and Avci (2019), we use the attribution prior loss, which calculates attribution values using integrated gradients (Sundararajan et al., 2017). We define c as the total number of intent classes. Each text utterance X_i consists of m_i tokens. Each token is marked with binary explanation signal t_i. We make slight variations in the original methodology. Liu and Avci (2019) uses the attribution using the derivative of the output concerning only one specific class, but following Jayaram and Allaway (2021), we sum over the attributions of all the classes as specified in equation 1 for regularization and use its average in equation 2. We compute the average across output classes as the impact of every neuron, whether positive or negative, is integral to the model's decision (Jayaram and Allaway, 2021).

$$a_i = \sum_{j=1}^{c} a_i^j / c \qquad (1)$$

$$L^{prior}(a, t) = \sum_{i=1}^{n} (a_i - t_i)^2 \qquad (2)$$

This attribution loss is added to the regular cross-entropy loss function used for simple classification.

$$L^{joint} = L(y, p) + \lambda \sum_{i=1}^{n} (a_i - t_i)^2 \qquad (3)$$

$$L(y, p) = \sum_{c}^{C} -y_c * log(y_c) \qquad (4)$$

Unlike Liu and Avci (2019), we do not penalize the model based on a universal token list. Instead, we use local explanation signals as each sample's target values. Hence, the value of $t_i \in \{0, 1\}$

within each sample X_i serves as its respective set of target values. This is done because a word that is an explanation signal for one sample may not be an explanation signal to another.

5.2 Experiment details

We fine tune CNN (Kim, 2014), LSTM (Hochreiter and Schmidhuber, 1997), bert-base-uncased (BERT) (Devlin et al., 2019), RoBERTa-base (RoBERTa) (Liu et al., 2019), GPT-2 (Radford et al., 2019) models on both ATIS and SNIPS datasets. All the models are evaluated on both classification and explainability metrics. We calculate attribution scores using integrated gradients and LIME as it performs better for explainability properties than other post-hoc techniques (Atanasova et al., 2020). For CNN, LSTM, BERT, and RoBERTa, we evaluate performance on explainability metrics with simple model training and models trained with joint attribution loss as stated in equation 3. For training with joint loss, we guide the model to focus on silver annotated explanation signals. Attribution scores are calculated using integrated gradients for joint training. We maintain a constant number of 50 steps to calculate attribution scores using integrated gradients. Other hyperparameter details and model architecture details are provided in appendix A

6 Evaluation

6.1 Evaluation metrics

An accuracy metric is used to evaluate the model's predictability. We evaluate the explainability of all the models using *plausibility* and *faithfulness* metrics (DeYoung et al., 2019). *Plausibility* compares the model's reasoning with the ground truth explanation. Here, we use a silver annotated explanation as ground truth. *Faithfulness* serves as a metric for evaluating the degree to which the model's provided explanation aligns with its actual reasoning.

We report the *Token F1 score* and *IOU F1* scores for plausibility. Token F1 score measures the mean F1 score of a direct match between rationale derived from the model and ground truth explanation signals. IOU F1 score allows credit for partial matches. Intersection over Union (IOU) metric at the token level is defined as the measure of token overlap between two spans divided by the size of their combined token sets. It considers a prediction a match if it intersects with any

ground truth rationales by more than 0.5. These partial matches are later used to compute the F1 score (DeYoung et al., 2019). We evaluate using the top 5 words with the highest attribution scores, as ground truth explanations typically consist of 4-5 words. For *faithfulness* we evaluate models on *Comprehensiveness* and *Sufficiency* metrics (DeYoung et al., 2019)

Sufficiency evaluates the extent to which the extracted rationales provide the necessary information for a model to make decisions. If model m predicts class j for text sample x_i, then the probability of that prediction is denoted by $m(x_i)_j$. r_i denotes the predicted rationale. Sufficiency is measured as Sufficiency = $m(x_i)_j - m(r_i)_j$.

Comprehensiveness measures the importance of the predicted rationale by evaluating the change in prediction when the rationale is removed from the text utterance. It is defined as:

$$\text{Comprehensiveness} = m(x_i)_j - m(x_i \setminus r_i)_j \quad (5)$$

where $m(x_i \setminus r_i)_j$ represents the probability of predicting class j when the rationale r_i is removed from the text sample x_i..

6.2 Results

We find that models perform well on intent classification but do not perform well on explainability metrics *plausibility* and *faithfulness*. We observe that improving the reasoning of models using regularization helps improve the model's predictions for the classes with scarce data. Training with attribution loss results in a 3-4% improvement in classification accuracy for both CNN and LSTM models and a slight improvement for the BERT and RoBERTa models on the ATIS dataset. (refer to table 2). In particular, we observe that the model's prediction improves for classes with few training samples. For example, *atis-meal, atis-ground-fare, and atis-flight-no* etc. We believe there is less improvement in BERT and RoBERTa because of the strong priors obtained during pre-training. Besides prediction, training models on silver annotated explanations improve the models' performance in explainability metrics. It increases the token F1 score by 3-4% (Table 2 and Table 3) for both datasets. In both datasets, BERT and RoBERTa are observed to make more human-like decisions. As a result, they have significantly higher token F1 scores than CNN and LSTM. For the ATIS dataset, performance in all explain-

Dataset	Model	Accuracy	Token F1	IOU F1	Comprehensiveness	Sufficiency
ATIS	CNN	82.32%	0.232	0.06	0.38	**-0.068**
	CNN Joint	86.24%	0.261	0.06	0.34	-0.34
	LSTM	93.23%	0.234	0.07	0.42	0.001
	LSTM Joint	95.27%	0.266	0.08	0.45	-0.03
	BERT	97.2%	0.315	0.11	0.49	0.11
	BERT Joint	97.6%	0.346	0.12	0.45	0.06
	RoBERTa	97.6%	0.323	0.11	**0.49**	0.15
	RoBERTa Joint	**97.8%**	**0.354**	**0.12**	0.44	0.09
	GPT-2	88.51%	0.234	0.06	0.43	0.003
	GPT-2 Joint	91.63%	0.267	0.09	0.46	-0.06
SNIPS	CNN	97.14%	0.342	0.14	0.58	**-0.075**
	CNN Joint	97.17%	0.366	0.14	0.52	-0.022
	LSTM	97.83%	0.347	0.14	0.54	0.004
	LSTM Joint	97.86%	0.349	0.14	0.52	0.001
	BERT	98.31%	0.412	0.17	0.68	0.008
	BERT Joint	98.38%	0.432	0.17	0.63	0.004
	RoBERTa	98.51%	0.413	0.17	0.67	0.01
	RoBERTa Joint	**98.57%**	**0.426**	**0.18**	0.63	0.008
	GPT-2	98.44%	0.402	0.17	**0.69**	0.007
	GPT-2 Joint	98.47%	0.433	0.18	0.68	0.001

Table 2: Model performance results across different metrics. We use the integrated gradient to obtain the model's explanation. Model names with Joint keyword are trained with joint feature attribution loss. While, Token F1 and IOU F1 score denotes *plausibility*, Comprehensiveness and Sufficiency denotes *faithfulness*.

ability metrics drops significantly. A large number of fine-grained classes with subtle differences in meaning might be the reason for that. Even though CNN and LSTM struggle with the plausibility metric, they perform well in the sufficiency metric (table 2 and 3). We also observe that joint training the performance of models on the faithfulness metric. The LIME explanation has better faithfulness than gradient-based explanations (Table 3). This suggests the independence of all the explainability metrics as depicted by (DeYoung et al., 2019) and (Atanasova et al., 2020).

7 Analysis

We analyze the reasoning of models listed in table 2 using both LIME and integrated gradient explanations. It is observed that most of the classification is based on highly frequent words. Hence, models do not give good F1 scores for skewed datasets like ATIS (Table 4). For example, in the ATIS dataset, text utterance *At the Charlotte airport, how many different types of aircraft are there for us.* has gold intent *atis-aircraft*. To explain gold intent, the *aircraft* word must be present as an explanation. But BERT fine-tuned on ATIS

dataset relies on word *airport* as class *atis-airport* has more training samples. Training it on silver annotations from our dataset improves the model's reasoning and helps to predict the correct class. Although the model focuses on the most frequent terms a lot, it understands the role of the main predicate well. As a result, the explanation of 70-80% text utterances consists of the main predicate in words with top 5 attribution scores in all experiments. In the SNIPS dataset, all the models perform very well but struggle with explainability. For example, the class *SearchScreeningEvent* model should focus on the words "movies," "Play" and "theater." However, it focuses on terms like names of the movies or actors. Due to abundant samples, deep learning models perform well on the SNIPS dataset but might not generalize well across multiple domains in scarce data scenarios.

8 Conclusion, and Future work

We introduce a novel silver annotation technique using the main predicate and arguments to annotate word-level explanation signals automatically for intent classification. Using this technique, we augment standard intent classification datasets

ATIS and SNIPS. Hence, we contribute a first-of-its-kind benchmark dataset for explainability in intent classification consisting of 21k samples. We validate the quality of silver standard annotated explanations by detailed human evaluation. This approach can be adopted across diverse domains and intents. We evaluated several models on this dataset and found that models performing well on intent classification do not perform well in plausibility and faithfulness in explainability metrics. However, ingesting feature attribution priors improves the model's reasoning and overall performance.

In future work, one can attempt to improve the model's performance under scarce data scenarios by training it on explanation signals. It will be interesting to apply the concept of main predicate and arguments to other languages and build a multilingual benchmark dataset for explainability in intent classification.

Limitations

We do not leverage ground truth intent to generate explanations. Although slots in original cases are not explanation signals in most cases, they provide some hint of gold intent in some samples. Identification of such cases can help to improve explanations. Since our technique uses knowledge and linguistic information like Parts of speech and dependency relations, it may not work best with multilingual settings because language structure differs for different languages.

Ethics Statement

We use all the open-source datasets, and their use does not harm people's privacy using chatbots or any conversational AI platforms. All the models are trained on the same datasets; hence, it maintains people's privacy.

References

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020. A diagnostic study of explainability techniques for text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3256–3274, Online. Association for Computational Linguistics.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. 2020. Effi-

cient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 38–45, Online. Association for Computational Linguistics.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. Bert for joint intent classification and slot filling. *arXiv preprint arXiv:1902.10909*.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. 2019. Eraser: A benchmark to evaluate rationalized nlp models. *arXiv preprint arXiv:1911.03429*.

Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. 2021. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175.

Kalpa Gunaratna, Vijay Srinivasan, Akhila Yerukola, and Hongxia Jin. 2022. Explainable slot type attentions to improve joint intent detection and slot filling. *arXiv preprint arXiv:2210.10227*.

Shirley Anugrah Hayati, Dongyeop Kang, and Lyle Ungar. 2021. Does BERT learn as humans perceive? understanding linguistic styles through lexica. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6323–6331, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Charles T. Hemphill, John J. Godfrey, and George R. Doddington. 1990. The ATIS spoken language systems pilot corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27,1990*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Sahil Jayaram and Emily Allaway. 2021. Human rationales as attribution priors for explainable stance detection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5540–5554, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ratnesh Joshi, Arindam Chatterjee, and Asif Ekbal. 2021. Towards explainable dialogue system: Explaining intent classification using saliency techniques. In *Proceedings of the 18th International Conference on Natural Language Processing (ICON)*, pages 120–127.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Shiyang Li, Jianshu Chen, Yelong Shen, Zhiyu Chen, Xinlu Zhang, Zekun Li, Hong Wang, Jing Qian, Baolin Peng, Yi Mao, et al. 2022. Explanations from large language models make small reasoners better. *arXiv preprint arXiv:2210.06726*.

Frederick Liu and Besim Avci. 2019. Incorporating priors with feature attribution on text classification. *arXiv preprint arXiv:1906.08286*.

Yinhan Liu, Myle Ott, 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. *arXiv preprint arXiv:1907.11692*.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2021. Hatexplain: A benchmark dataset for explainable hate speech detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 14867–14875.

Grégoire Montavon, Alexander Binder, Sebastian Lapuschkin, Wojciech Samek, and Klaus-Robert Müller. 2019. Layer-wise relevance propagation: an overview. *Explainable AI: interpreting, explaining and visualizing deep learning*, pages 193–209.

Libo Qin, Tailu Liu, Wanxiang Che, Bingbing Kang, Sendong Zhao, and Ting Liu. 2021. A co-interactive transformer for joint slot filling and intent detection. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8193–8197. IEEE.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Christian Raymond and Giuseppe Riccardi. 2007. Generative and discriminative algorithms for spoken language understanding. In *Interspeech 2007-8th Annual Conference of the International Speech Communication Association*.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Anchors: High-precision model-agnostic explanations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615, Austin, Texas. Association for Computational Linguistics.

Zhengxuan Wu and Desmond C Ong. 2021. On explaining your explanations of bert: An empirical study with sequence classification. *arXiv preprint arXiv:2101.00196*.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2048–2057, Lille, France. PMLR.

Xi Ye and Greg Durrett. 2022. The unreliability of explanations in few-shot prompting for textual reasoning. *Advances in neural information processing systems*, 35:30378–30392.

Ruiqi Zhong, Steven Shao, and Kathleen McKeown. 2019. Fine-grained sentiment analysis with faithful attention. *arXiv preprint arXiv:1908.06870*.

A Appendix

A.1 Model details

A.1.1 CNN

We build a simple text classifier CNN model. The architecture consists of 256 filters, 2,3 and 4-gram. The network takes into input sequence length of 80. Following convolution layers, architecture consists of max-pooling and softmax function layers. The network performs well with the SNIPS

dataset and has decent performance for the ATIS dataset. In addition to the basic network, we ingest feature attribution during training to provide supervision over rationale. These feature attributions are calculated using an integrated gradient technique.

A.1.2   LSTM

For the LSTM model, a sequence of length 80 is passed as embeddings. The network consists of 2 fully connected layers of neurons with 32 and 32 neurons each. We also added a dropout layer to avoid overfitting. Like the CNN model, LSTM is also trained on joint attribution loss.

A.2   Hyperparameters details

All the training has been conducted on NVIDIA A100-SXM4-80GB GPU. We use a batch size of 32 for all the experiments. We use the cross-entropy loss function and the Adam optimizer for training. For the simple fine-tuning, the value of λ is kept at 0. To fix the value of hyperparameter λ, we extensively search the range values $(1, 10^{10})$ and select the values for which the best results are obtained. For joint attribution training of CNN and LSTM, λ is set to 10^5 and 10^4 for the SNIPS dataset. For the ATIS dataset, λ values are set as 10^6 for both CNN and LSTM models. The initial learning rate for BERT is 4e-5, and 0.001 for both CNN and LSTM. BERT was trained for 2, CNN for 10, and LSTM for five epochs for both datasets. All the training details for the Roberta and GPT-2 model are the same as those for BERT.

Dataset	Model	Token F1	IOU F1	Comprehensiveness	Sufficiency
ATIS	CNN	0.196	0.03	0.42	**-0.072**
	CNN Joint	0.228	0.03	0.38	-0.37
	LSTM	0.182	0.05	0.44	-0.06
	LSTM Joint	0.203	0.05	0.47	-0.09
	BERT	0.26	0.09	**0.53**	0.09
	BERT Joint	0.298	0.1	0.49	0.05
	RoBERTa	0.253	0.09	0.51	0.1
	RoBERTa Joint	**0.302**	**0.11**	0.49	0.06
SNIPS	CNN	0.33	0.14	0.62	**-0.081**
	CNN Joint	0.36	0.14	0.55	-0.085
	LSTM	0.33	0.14	0.55	0.001
	LSTM Joint	0.34	0.14	0.57	-0.42
	BERT	0.37	0.16	**0.72**	0.005
	BERT Joint	**0.4**	**0.17**	0.69	0.002
	RoBERTa	0.36	0.16	0.7	0.2
	RoBERTa Joint	0.39	0.17	0.63	0.03

Table 3: Model performance results across different metrics. Here, LIME is used to obtain the model's explanation. Model names with Joint keyword are trained with joint feature attribution loss

Class names	precision	recall	F1 score	samples
atis-abbreviation	0.94	1.0	0.97	33
atis-aircraft	0.8	0.89	0.84	9
atis-airfare	0.77	1.00	0.87	48
atis-airline	0.9	1.0	0.95	38
atis-airport	0.95	1.0	0.97	18
atis-capacity	1.00	0.52	0.69	21
atis-city	1.00	0.67	0.8	6
atis-distance	0.77	1.00	0.87	10
atis-flight	0.99	0.99	0.99	632
atis-flight-no	1.00	0.38	0.55	8
atis-flight-time	0.5	1.00	0.67	1
atis-ground-fare	1.00	0.57	0.73	7
atis-ground-service	0.92	1.00	0.96	36
atis-meal	1.00	0.33	0.5	6
atis-quantity	0.25	1.00	0.4	3

Table 4: BERT model class-wise performance when fine-tuned on ATIS dataset. Column samples denote the number of samples for that class.