# ARISE: Automatic Rule Induction and Filtering for Few-shot Text Classification

**Anonymous ACL submission**


## Abstract

We propose ARISE, a framework that combines weak supervision, synthetic data generation, and contrastive representation learning for few-shot text classification (FSTC). Weak supervision forms a major novelty in ARISE. Here, we propose an automatic rule induction component to induce rules from syntactic-ngrams using inductive generalisation. The rules we induce capture syntactic information, often not explicitly captured by state-of-the-art neural models. While these rules can be noisy, they are used to learn a label aggregation model with data programming. Subsequently, we jointly train the base classifier along with the label aggregation model to update their parameters. Unlike, past work that employ data programming to label unlabeled data points, we use it for verifying synthetically generated labeled data. Finally, we combine synthetic data generation and automatic rule induction, via bootstrapping, to iteratively filter the generated rules and data. Our experiments with nine FSTC datasets over diverse domains, and multilingual experiments on seven languages, show consistent and statistically significant improvements for our proposed approach over other state-of-the-art approaches.


## 1 Introduction

Few-shot text classification (FSTC) is challenging, especially in tasks with a large, semantically similar and often overlapping label space (Zhang et al., 2022b). Such tasks often find application in diverse domains including task oriented dialogue (intent classification), e-commerce, social networks, scientific literature etc. (Yehudai and Bendel, 2024; Zhang et al., 2021b). Moreover, these tasks are expected to have a unique or highly specialized label space, leading to limited availability of annotated data (Singhal et al., 2023; Vulić et al., 2022). Intuitively, FSTC systems should be designed to extract as much information as possible from the limited supervision data available for learning. We propose ARISE, a framework that combines automatic rule induction (Pryzant et al., 2022; Bajpai et al., 2024), synthetic data generation, and contrastive representation learning (Zhang et al., 2022b) for FSTC. Moreover, ARISE induces rules in the form of syntactic n-grams that complements information captured in prevalent approaches in FSTC.

FSTC tasks are generally addressed using a diverse set of techniques. These include In-context learning (Brown et al., 2020; Kojima et al., 2022), contrastive representation learning (Vulić et al., 2021), data augmentation and filtering (Lin et al., 2023), transductive learning (Singhal et al., 2023), weak supervision (Pryzant et al., 2022), meta-learning (Mesgar et al., 2023) among others. Several of these works successfully combine one or more of these techniques for FSTC tasks (Singhal et al., 2023; Vulić et al., 2022).

In ARISE, we propose a bootstrapped approach for iterative synthetic data generation and automatic rule induction (Yarowsky, 1995; Varma and Ré, 2018). Moreover, it enables joint training of the induced rules with pre-trained neural models via data programming (Maheshwari et al., 2021; Zhang et al., 2022a). Figure 1 shows various components and the 3-step workflow for ARISE. One, our rule induction step extracts syntactic ngrams from sentence-level dependency parses of the labeled input. Rules are induced from the syntactic n-grams via inductive generalization using least general generalization (LGG Plotkin, 1971; Raza et al., 2014). The induced rules are then filtered using a submodular graph cut-based function (Bajpai et al., 2024; Kothawade et al., 2022). Two, the data augmentation step, involves synthetic generation of data using in-context learning (Liu et al., 2022). Synthetic data are generated along with their labels, which are then validated using the rules. Only those labeled data points that match with the predictions of the rules are filtered. Iteratively, we induce

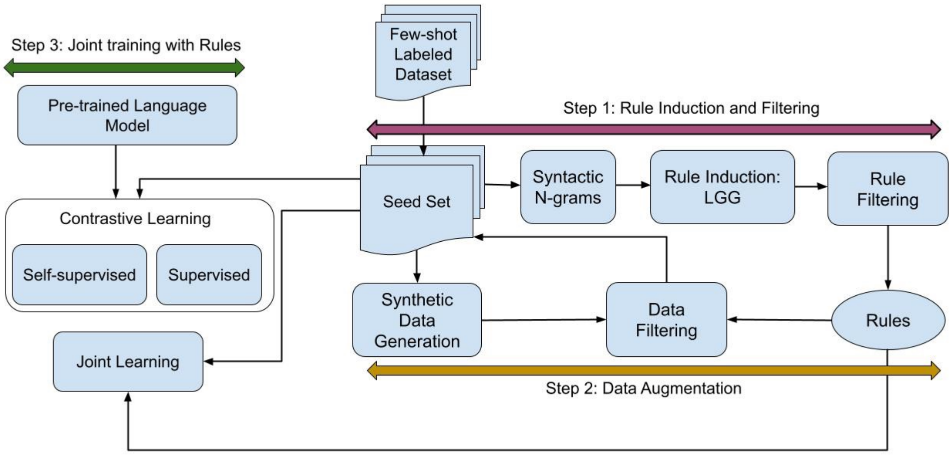


Figure 1: Three-step workflow for ARISE, along with various components in it.

rules from synthetically generated data and use the induced rules for data filtering.

Three, the joint learning step, effectively combines contrastive representation learning, (Chen et al., 2020; Khosla et al., 2020a) supervised fine-tuning, and Data programming (Zhang et al., 2022a) using a joint learning framework (Maheshwari et al., 2021). We perform self-supervised contrastive pretraining (Wu et al., 2020) and supervised contrastive learning (Khosla et al., 2020b; Zhang et al., 2022b) over a standard pre-trained neural classifier. We use the few-shot labeled data, along with the filtered data, for fine tuning the neural classifier. The induced rules enable learning a generative model as a form of weak supervision using data programming. We jointly learn a classifier with the generative model using SPEAR Maheshwari et al. (2021), a data programming framework.

In ARISE, we induce generalized syntactic n-grams as our rules. Our primary aim here is to potentially capture morpho-syntactic information from data, which currently is not captured explicitly by other learning techniques and models employed in ARISE. A classical NLP pipeline typically represents a string at multiple levels of abstraction which includes POS tags, syntactic relations, *etc.* (Manning et al., 2014). ARISE use higher-order dependency structures as features and generalize over these features using inductive generalization (Popplestone, 1970) to induce the rules as generalized syntactic n-grams.

We perform extensive experiments on the 'Few-Many' benchmark (Yehudai and Bendel, 2024), consisting of eight datasets for a diverse set of FSTC tasks. We additionally include experiments for the 'SciCite' (Cohan et al., 2019) dataset, a dataset from the scientific literature domain. Further, we perform multilingual experiments on seven languages using the MASSIVE dataset. Our experiments are performed using both 5-shot and 10-shot settings. In all these settings, ARISE outperforms strong competitive models, such as IntenDD (Singhal et al., 2023), Zhang et al. (2022b), and FastFit (Yehudai and Bendel, 2024), with statistically significant improvements.

In section 2, we elaborate on our rule induction approach for inducing generalized syntactic n-grams. In section 3, we elaborate ARISE, a 3-step framework for FSTC. Here, we elaborate our iterative rule and data filtering along with the joint learning setup.

Our major contributions are as follows:

- Our proposed approach yields statistically significant gains in all the experiments we perform, compared to state-of-the-art systems (Yehudai and Bendel, 2024; Singhal et al., 2023; Zhang et al., 2022b). Our best performing model reports a 2.04 % increase in 10-shot and 2.52 % increase in 5-shot settings, compared to the next best model, averaged across all the monolingual tasks.

- Our extensive experiments show that ARISE is generalizable and across multiple domains (as reported above) and multiple languages. We report a 4.4 % increase in performance, compared to the next model, averaged across seven different languages.

- We show that leveraging syntactic information as weak supervision for rule induction, leads to performance improvements compared to surface-level string n-grams as rules. Further, our bootstrapped approach outperforms competitive approaches for filtering augmented data (Lin et al., 2023).

## 2 Automatic Rule Induction Using Syntactic Tree Generalization

Distributional hypothesis (Firth, 1957) is often realized using vector space models defined over a suitable feature space (Turney and Pantel, 2010). Inputs can be encoded into a feature space of dense contextualized vectors (Peters et al., 2018; Devlin et al., 2019) or into a sparse semantic space consisting of lexical n-grams, syntactic n-grams (Goldberg and Orwant, 2013), higher order dependency features (Koo and Collins, 2010), or even graph motifs (Biemann et al., 2016).

We induce rules that can capture complementary information that is not explicitly captured in pre-trained neural models. Hence, we focus on incorporating structured grammatical information typically used in a traditional NLP pipeline (Manning et al., 2014) such as Part-of-Speech (PoS) and syntactic information. From dependency parses of input sentences, we extract induced subtrees as features. Each such feature is a syntactic n-gram, with the nodes as the words and the edges labeled with the dependency relations. We then induce rules via the inductive generalization of these features, using Least General Generalization (Raza et al., 2014; Thakoor et al., 2018).

For an FSTC task with $k$ labels, we assume the availability of few-shot labeled dataset $\mathcal{D}$, where $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}$, $x_i$ is an input document and $y_i \in \{l_1, l_2, ..., l_k\}$ is a label. We obtain sentence-level dependency parses for each $x_i \in \mathcal{D}$. A feature space $\mathcal{F}_{t=1}^{f}$ is defined over higher-order factorization of the dependency parses in $\mathcal{D}$. Each feature $f_t \in \mathcal{F}$ is an induced subtree of the parses for sentences in $\mathcal{D}$. Here, a feature covers a set of documents in which that feature occurs at least once.

Rules are generalizations of features. If a generalized rule subsumes multiple features, then it covers a union of all the sets of documents corresponding to those features. The rules we generate belong to $\mathcal{R}_{t=1}^{r}$, where for every input in $x_i \in \mathcal{D}$ it should either predict a label from $\{1, ..., k\}$ or should abstain $(-1)$ from making a prediction. Our rules are induced as the least general generalization (LGG) over a set of features (Plotkin, 1970, 1971). A feature can be a rule in itself, i.e. $\mathcal{F} \subseteq \mathcal{R}$. For forming the rules we define two forms of generalizations, structural and linguistic. if rule $r_i$ is an induced subtree of $r_j$, then we can say that $r_i$ is more general than $r_j$. Linguistic generalization include, substitution (Raza et al., 2014; Thakoor et al., 2018), of the nodes containing words with their corresponding stems, and PoS tags (Galitsky and Ilvovsky, 2019).

Figure 2 shows illustrative cases of generalization. Let us consider a corpus from which features (syntactic n-grams) $f_1$ to $f_6$ are extracted. Now, $r_1$ to $r_8$ shows various generalized rules induced from these features. Rules $r_1$ to $r_7$ show linguistic generalization and $r_8$ shows structural generalization (from $r_7$). Consider rules $r_1, r_4, r_5$ and $r_7$. These rules contain nodes with a group of words. Similarly, $r_6$ represents a rule that has a group of PoS tags in one of the nodes. In linguistic generalization, multiple trees are generalized to a single tree by grouping words or PoS that differ in these individual trees. Here, $r_1$ is a generalisation of $f1$ and $f_2$. Similarly, $r_4$ is a generalization of $f_2$ and $f_3$. Currently, we restrict the groupings at a node to be homogeneously typed, i.e. a set can either be that of inflected word forms, stems or of PoS tags, but not a mix of those. Further, the cardinality of such a group is set to an arbitrary upper bound, to avoid trivial generalisations.

### 2.1 Rule Induction via LGG

We obtain features from dependency parses of the dataset $\mathcal{D}$. We consider only those subtrees that exactly have one of the six core dependency relations in them (de Marneffe et al., 2014; Nivre et al., 2020). These core dependency relations are direct or indirect object, nominal or clausal subject, clausal complement or open clausal complement. We partition the features into 6 mutually exclusive subsets, one each for each of the core relations.

A complete lattice is constructed out of each partition, by adding a supremum and infimum element to the partition. Here, we add a rule '$* \xleftarrow{rel} *$', where 'rel' is the core-relation corresponding to the partition. It is the supremum for any element in the partition, as every element in the partition is subsumed by it and covers any document that has the relation present in it. We also define '$\epsilon$' as the infimum and it represents an empty rule that rules out any document in the input. The complete lattice provides a search space of rules over which the partial ordering is provided. Here, any two pair of subtrees have a least general generalization or a least upper bound (Raedt, 2010). In Figure 2, $r_1$ is the LGG of $f_1$ and $f_2$. $r_1$ represents all the sentences that either have $f_1$ or $f_2$ in their dependency parses. Similarly, $r_2$ and $r_3$ are also generalizations

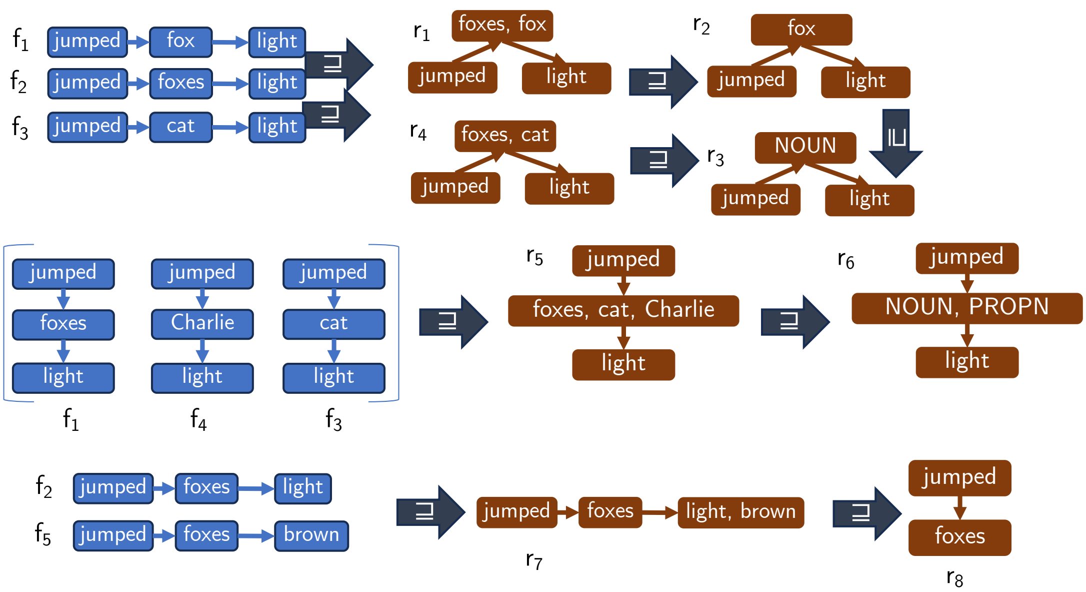


Figure 2: Rule induction from syntactic n-grams via inductive generalization. The symbol '$\sqsupseteq$' denote a generalization operation. Trees labeled from $f_1$ to $f_6$ are instances of features, and the nodes of these trees are colored using . Similarly, trees labeled from $r_1$ to $r_8$ are rules and the nodes of these trees are colored using .

of $f_1$ and $f_2$, but not their LGG.

For every rule in the lattice, we compute its label-PMI vector, following Singhal et al. (2023) and Jin et al. (2022). label-PMI vector is a vector of the pointwise mutual information scores of the rule corresponding to each label. From the vector, we consider its maximum score, denoted as L-PMI. The label corresponding to L-PMI is then assigned to the rule. From the lattice, we start bottom up and compute the LGG for every pair of rules. We induce the LGG as a rule, only if it has a higher L-PMI than the individual rules in the pair. The rules thus induced form our candidate set of rules.

## 3 ARISE Framework

### 3.1 Rule Induction and Filtering

We induce rules from a set of input documents (§2). These rules can be used as labeling functions in Data Programming, henceforth to be referred to as Programmatic Weak Supervision (PWS), for learning a generative model (Ratner et al., 2017; Zhang et al., 2022a). While the individual rules are expected to be noisy in PWS, the final set of filtered rules needs to be accurate, diverse and high in coverage (Bajpai et al., 2024).

For rule filtering, we use the submodular graph-cut (GC) function (Kothawade et al., 2022), as proposed by Bajpai et al. (2024). Using GC, we select a final set of representative and diverse rules $\mathcal{R}_f$, from the set of candidate rules $\mathcal{R}$. For $\mathcal{R}_f \subseteq \mathcal{R}$, we define the GC function as $f_{GC}(\mathcal{R}_f) = \sum_{r_i \in \mathcal{R}, r_j \in \mathcal{R}_f} s_{ij} - \lambda \sum_{r_i, r_j \in \mathcal{R}_f} s_{ij}$. Here, $\lambda \in [0, 1]$ governs the diversity-representation trade-off, where higher $\lambda$ implies higher diversity in $\mathcal{R}_f$. $s_{ij}$ is the similarity score for rule pair $r_i$ and $r_j$. It is calculated as the weighted sum of the precision, coverage, and agreement between the pair of rules, $s_{ij} = \alpha(r_i) + \alpha(r_j) + w * \beta(\{r_i, r_j\}) + \gamma * \mu(r_i, r_j)$. Here, $\alpha(r_i) = \text{Precision}(r_i)$, $\mu(r_i, r_j)$ is the agreement, calculated as the fraction of instances where both rules agree. $\beta(\{r_i, r_j\})$ is the coverage, calculated as the fraction of instances labeled by at least one of the rules.

Our objective function is $\max_{|\mathcal{R}_f| \leq k} f_{GC}(\mathcal{R}_f)$, where $k$ is a fixed budget (Kothawade et al., 2022). We greedily choose a rule that maximizes the marginal utility $argmax_{r_i \in \{\mathcal{R} - \mathcal{R}_f\}} f_{GC}(\mathcal{R}_f \cup \{i\}) - f_{GC}(\mathcal{R}_f)$. Please note that (Bajpai et al., 2024) starts from an empty set, while we start with the existing rule set obtained from the previous round of bootstrapping. One round of filtering is completed until the fixed budget $k$ is completed.

### 3.2 Bootstrapping Rules and Synthetic Data

PWS is typically employed to provide noisy training labels to unlabeled data (Varma and Ré, 2018). In ARISE, we instead use PWS on synthetically generated data for data verification and filtering.

We start our bootstrapping with the few-shot gold labeled data as the seed, as shown in Figure 1.

we synthetically generate new data for each class using the few-shot prompt demonstrations, with the demonstrations retrieved from the seed set. Zhu et al. (2023) observe that PWS systems rely heavily on the quality of gold-labeled data, especially in the validation split. Hence, we use our gold-labeled data as validation data for rule filtering. We perform rule induction (§2) from the synthetically generated data and filter the rules (§3.1) using the gold data as the validation split. The induced rules are then used for learning a generative model via PWS (Chatterjee et al., 2020). Finally, the seed set is expanded with filtered data, where only those data points that match their generated label with the predicted label from the generative model are filtered. Our validation set is never expanded and is always the gold-labeled data. The seed data set is expanded with newly filtered data after every iteration. Similarly, the rule set is also expanded after every iteration of the bootstrapping process.

**Data Augmentation:** We use few-shot prompt demonstration to synthetically generate new labeled sentences using LLMs. For each label, we sample k instances each of positive and negative samples from the seed set and then use it for generating new data samples (Smith et al., 2024; Lin et al., 2023). Our prompt demonstration approach includes label information, positive examples, and negative examples for synthetic generation. In addition to generating new data points, we also perform paraphrasing of data points in the seed set. By paraphrasing, we gain diverse syntactic structures for better rule induction.

### 3.3 Joint Learning with Rules

The few-shot classifier is trained using SPEAR (Maheshwari et al., 2021), a joint learning framework that learns over a feature-based classification model and a label aggregation (LA) model. The feature model is a pre-trained neural model and LA is a generative model (Chatterjee et al., 2020), learned via PWS, using the automatically induced rules as labeling functions. LA is denoted as $P_\theta(\mathbf{l}_i, y)$, where $\mathbf{l}_i$ a vector that represents the firing of all LFs for an input $\mathbf{x}_i$. Each firing, $l_{ij}$ can be either 0 (abstain) or class label $k$ (Chatterjee et al., 2020).

Following Maheshwari et al. (2021), our joint learning objective incorporates three different loss components for learning from labeled data. We provide a brief overview of each loss component below, while encouraging interested readers to (Maheshwari et al., 2021) for detailed information.

$$\min_{\theta,\phi} \sum_{i\in\mathcal{L}} L_{CE}\left(P_\phi^f(y|\mathbf{x}_i), y_i\right) + LL_s(\theta|\mathcal{L}) \\ + \sum_{i\in\mathcal{L}} KL\left(P_\phi^f(y|\mathbf{x}_i), P_\theta(y|\mathbf{l}_i)\right)$$

The first component of the loss is the standard cross-entropy loss for the model $P_\phi^f$. The second component is the negative log-likelihood on the dataset. The third is the KL-Divergence between the predictions of the LA and $P_\phi^f$ models, which enforces consensus by aligning their predictions.

**Contrastive Representation Learning:** The pre-trained model, $P_\phi^f$, undergoes contrastive representation learning prior to joint learning. Following, Zhang et al. (2021a) and Singhal et al. (2023), we first perform self-supervised contrastive learning (SSCL) over a pre-trained model. Here, the model parameters for a given pre-trained model is updated using, $\mathcal{L}_{pt} = \mathcal{L}_{sscl} + \lambda_{pt}\mathcal{L}_{mlm}$. $\mathcal{L}_{pt}$ is a weighted sum of token-level masked language modeling loss ($\mathcal{L}_{mlm}$) and a sentence-level SSCL ($\mathcal{L}_{sscl}$; Wu et al., 2020; Liu et al., 2021). $\lambda_{pt}$ is a weight hyper-parameter. For SSCL, given an input document $x_i$, we obtain perturbations of $x_i$ by randomly masking tokens from it. Further, we dynamically mask tokens such that each sentence has different masked positions across different training epochs. SSCL attempts to bring the $x_i$ and its masked versions closer in the semantic space while pulling away other pairs.

After the continued pretraining, we perform supervised contrastive learning (Khosla et al., 2020a). Here, we try to increase the similarity between input pairs that belong the same class, while trying to bring down the similarity of those belonging to different classes. We follow the supervised contrastive learning (Khosla et al., 2020a) loss, where all the documents in the same class in a batch are brought together. Here, the same document may also be used to create like pairs by creating perturbations of the input.

## 4 Experiments

**Dataset** : We use FewMany Benchmark (Yehudai and Bendel, 2024), for our monolingual experiments. FewMany consists of eight FSTC datasets (Yehudai and Bendel, 2024). It consists of CLINC150 (C150; Larson et al., 2019), Banking77 (B77; Casanueva et al., 2020), HWU64

| | FT | CL | DA | Filtering for DA | | | Automatic Rule Induction | | IDRF | ICL |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | PVI | ST | GC | ngrams | syntactic ngrams | | |
| Base | ✓ | | | | | | | | | |
| Base-DA | ✓ | | ✓ | | | | | | | |
| Base-ST | ✓ | | ✓ | | ✓ | | | | | |
| IntenDD | ✓ | ✓ | ✓ | | | ✓ | ✓ | | | |
| Snorkel | ✓ | | ✓ | | | ✓ | | ✓ | | |
| CPFT | ✓ | ✓ | ✓ | ✓ | | | | | | |
| FastFit | ✓ | ✓ | ✓ | ✓ | | | | | | |
| CPFT + Snorkel | ✓ | ✓ | ✓ | | | ✓ | | ✓ | | |
| ARISE | ✓ | ✓ | ✓ | | | ✓ | | ✓ | | |
| ARISE-Iter | ✓ | ✓ | ✓ | | | ✓ | | ✓ | ✓ | |
| LLMs | | | ✓ | ✓ | | | | | | ✓ |

Table 1: Techniques used by competing systems. Base is Roberta and XLM-R for monolingual and multilingual experiments respectively. FT is fine-tuning; CL is contrastive learning; DA is data augmentation; PVI is pointwise V-information; ST is self-training; IDRF is Iterative Data and Rules filtering.

(HU64; Liu et al., 2019a) for intent classification; ARGUMENT TOPIC (AT71; Gretz et al., 2020) and CLAIM STANCE (CS55; Bar-Haim et al., 2017) for Topic classification; TREC question classification dataset (T50; Li and Roth, 2002), AMAZON PRODUCTS (AP106) and DBPEDIA (DB70). We also use SciCite (SC3 Cohan et al., 2019), from the scientific literature domain. Finally, the multilingual experiments are performed using the MASSIVE dataset (FitzGerald et al., 2023). Here, we use seven typologically diverse languages including Chinese, English, French, German, Hindi, Japanese, and Spanish.

**Data Augmentation:** We use GPT-3.5, 4, and Claude 3 Opus for synthetic data generation. We generate label-specific data by prompt demonstration. Here, Using Wu et al. (2023), we perform $k$-NN retrieval, with $k = 5$, from the seed data for positive demonstrations, and randomly sampled out of class samples as negative examples (Liu et al., 2022). For multilingual experiments, we experiment with *direct* generation of the synthetic data in the target language, and also via *translation* of synthetically generated English sentences. For translation, in addition to the three aforementioned LLMs we use NLLB-54B (Team et al., 2022) and Google Translate. For translation in Hindi, we use Gala et al. (2023).

**Baselines:** Table 1 shows our baselines. Base models are the 'Large' variants of Roberta (Liu et al., 2019b) and XLM-R (Conneau et al., 2020) for our monolingual and multilingual experiments respectively. Further, Base-DA is fine-tuned with augmented data (no filtering). Base-ST is trained using self-training-based filtering of augmented data. We also include competitive models that also combine multiple learning techniques, such as IntenDD (Singhal et al., 2023), Snorkel (Ratner et al., 2017), CPFT (Zhang et al., 2022b), and FastFit (Yehudai and Bendel, 2024). Following Yehudai and Bendel (2024), we report results for ICL, in 5-shot setups, using Flan-XXL (Wei et al., 2021), Flan-UL2 (Tay et al., 2022).

**Experimental Setup:** ARISE and ARISE-Iter, as shown in Table 1, are two variants without and with the iterative data and rule filtering (IDRF). ARISE variants use the same pre-trained models as used in 'Base'. We perform all our experiments using 5 random splits and report the average. We use accuracy as our metric and experiment with both 5-shot an 10-shot settings(Yehudai and Bendel, 2024). For joint learning, we use a 20 % split of the synthetically generated data as a validation split, while using all the gold data in training. For learning the parameters for our rule filtering step (§3.1), we use the few-shot gold data as validation. We report results for ARISE-iter induced with rules where the gold data was used only in the last iteration of bootstrapping. We keep a multiplier of 128x for our k-shot classification settings, following Lin et al. (2023). We use the graph-based

| Models | AP106 | AT71 | B77 | C150 | CS55 | DB70 | HU64 | T50 | SciCite | Avg |
|---|---|---|---|---|---|---|---|---|---|---|
| Base | 57.36 | 95.59 | 87.55 | 94.3 | 91.06 | 87.03 | 86.28 | 86.57 | 82.12 | 85.32 |
| Base-Aug | 57.42 | 95.3 | 88.36 | 93.83 | 90.16 | 87.92 | 87.58 | 86.8 | 82.58 | 85.55 |
| Base-ST | 58.46 | 95.78 | 88.58 | 94.37 | 91.1 | 88.23 | 88.69 | 87.26 | 83.07 | 86.17 |
| CPFT | 58.82 | 96.67 | 89.51 | 95.03 | 91.34 | 89.14 | 89.76 | 89.42 | 84.38 | 87.12 |
| Snorkel | 59.47 | 96.35 | 90.49 | 94.96 | 90.33 | 88.42 | 89.2 | 89.3 | 85.21 | 87.08 |
| FastFit | 59.29 | 96.79 | 89.4 | 95.48 | 90.24 | 88.63 | 89.54 | 88.84 | 85.01 | 87.02 |
| IntenDD | 59.67 | 97.02 | 90.07 | 95.71 | 91.71 | 88.93 | 89.04 | 88.45 | 85.04 | 87.29 |
| CPFT+ Snorkel | 59.74 | 97.12 | 90.76 | 95.24 | 91.48 | 89.22 | 89.81 | 89.71 | 85.67 | 87.64 |
| ARISE | 60.87* | 97.02 | 92.12* | 96.37* | 91.78 | 89.59 | 90.89* | 90.24 | 85.87 | 88.31 |
| **ARISE-Iter** | **62.6** | **97.93** | **92.82** | **97.15** | **92.89** | **90.78** | **92.27** | **91.32** | **87.12** | **89.43** |

Table 2: Accuracy Results for 10-shot monolingual FSTC. Results in boldface and those marked with * are statistically significant by t-test ($p < 0.05$) compared to ARISE and CPFT+Snorkel respectively.

biaffine parser (Dozat and Manning, 2016) trained with XLM-R as the encoder on the UD treebank (Zeman et al., 2023) for dependency parsing. We obtain induced subtrees of upto 3 nodes as rules.

### 4.1 Results

ARISE-iter, our proposed model, reports the best performance in all our experimental settings, as shown in Tables 2, 3, and 4. It outperforms all other models with statistically significant gains. ARISE-Iter reports an absolute improvement of 1.79 % points (2.04 % increase), 1.32 % and 2.58 % points averaged across the datasets, for the 5 and 10-shot monolingual and 10-shot multilingual setups.

### 4.2 Monolingual Results

ARISE-Iter and ARISE differs only in terms of bootstrapping (IDRF). Bootstrapping alone leads to an average absolute gain of 1.12 and 1.32 % points for the 10-shot and 5-shot setups respectively (Tables 2 and 3), between both the ARISE-Iter and ARISE respectively. Base-Aug reports statistically significant gains only for 3 of 9 datasets (B77, HU64, and DB70) compared to Base in Table 2. It shows that data augmentation without any filtering need not always improve the results. Further, Base-ST on average report a gain of 0.85 % points compared to Base, with statistically significant gains in 6 of 9 datasets (except for AT71, C150, and CS55).

ARISE variants follow CPFT (Zhang et al., 2022b) in employing contrastive learning (CL) components. CL components alone in CPFT lead to an average absolute gain of 1.8 % points compared to Base, in Table 2. Similarly, Snorkel, a PWS framework, and ARISE is trained with the same filtered data and rules. However, unlike ARISE, Snorkel does not use joint learning. Instead, Snorkel learns a generative model to label (or filter in our case) synthetically generated sentences. It outperforms the base model by an average absolute improvement of 1.76 % points and is competitive with CPFT. Snorkel and CPFT report statistically significant gains compared to Base for all the datasets, except CS55. Snorkel and CPFT report comparable performance on 5 of 9 datasets, with statistically significant gains in 2 datasets each.

CPFT+Snorkel combines both contrastive representation learning and PWS. It differs from ARISE, only in terms of the joint learning component. ARISE reports an absolute improvement of 0.67 % points in 10-shot settings (Table 2), and 0.96 % points in 5-shot settings (Table 3), as compared to CPFT+Snorkel. Results from Snorkel, CPFT+Snorkel, and ARISE show our rule induction component, as a general-purpose one for PWS. Similarly, gains in CPFT+Snorkel and ARISE show that combining complementary learning techniques leads to performance gains compared to using them independently.

ARISE-Iter, our proposed approach with IRDF, outperforms both IntenDD (Singhal et al., 2023) and FastFit (Yehudai and Bendel, 2024), two competitive models with state-of-the-art results on few-shot learning. While FastFit originally does not use data augmentation, we add augmented sentences to it for a fair comparison. IntenDD differs from ARISE by using string-level n-grams for weak supervision and additionally employs a two-level transductive learning approach. ARISE-Iter

| Methods | AT71 | B77 | C150 | CS55 | HU64 | T50 | Avg. |
|---|---|---|---|---|---|---|---|
| Flan-ul2 | 97.07 | 71.21 | 80.6 | 89.57 | 76.2 | 64.86 | 79.92 |
| Flan-XXL | 96.72 | 72.04 | 81.99 | 50.24 | 75.13 | 84.72 | 76.81 |
| Base | 95.61 | 79.77 | 91.67 | 87.94 | 79.29 | 73.67 | 84.66 |
| FastFit | 96.45 | 86.14 | 93.77 | 88.16 | 84.6 | 84.8 | 88.99 |
| Intendd | 96.11 | 89.13 | 94.05 | 88.76 | 88.21 | 86.86 | 90.52 |
| CPFT+ Snorkel | 96.74 | 88.64 | 94.46 | 88.57 | 87.38 | 87.45 | 90.54 |
| ARISE | 96.68 | 90.35 | 94.89 | 90.3 | 88.04 | 88.72 | 91.5 |
| **ARISE-Iter** | **97.14** | **91.68** | **96.13** | **91.59** | **90.22** | **90.14** | **92.82** |

Table 3: Accuracy Results for 5-shot monolingual FSTC.

| | En | De | Ja | Es | Fr | Zh | Hi | Avg. |
|---|---|---|---|---|---|---|---|---|
| Base | 77.65 | 71.23 | 74.89 | 71.56 | 72.81 | 73.14 | 71.07 | 73.19 |
| IntenDD | 79.55 | 73.64 | 76.5 | 76.92 | 76.42 | 76.53 | 74.41 | 76.28 |
| Snorkel | 80.52 | 75.39 | 78.87 | 75.79 | 77.65 | 76.7 | 74.16 | 77.01 |
| CPFT | 78.65 | 73.45 | 77.56 | 74.99 | 76.74 | 75.58 | 73.66 | 75.8 |
| FastFit | 80.73 | 75.97 | 78.49 | 75.64 | 76.84 | 75.98 | 74.07 | 76.82 |
| CPFT + Snorkel | 81.43 | 76.67 | 79.34 | 76.43 | 78.14 | 77.66 | 75.04 | 77.82 |
| ARISE | 82.43 | 76.64 | 79.52 | 77.1 | 78.93 | 78.32 | 75.16 | 78.3 |
| ARISE-Iter | **84.96** | **79.38** | **81.87** | **79.58** | **80.16** | **79.45** | **77.41** | **80.4** |

Table 4: Multilingual results on MASSIVE Dataset.

when trained with string level n-grams as used in IntenDD still outperforms IntenDD but reports an average accuracy of 88.64 %, a drop from 89.43 for the 10-shot setting. Similarly, the use of PVI for data filtering instead of IRDF for ARISE-Iter results in an average accuracy of 88.19 %.

Table 3 reports results for the 5-shot setup. We follow the setup of Yehudai and Bendel (2024) for ICL. ARISE-Iter reports an average absolute gain of 16.01 % and 2.28 % compared to Flan-XXL and CPFT+Snorkel models respectively. It also reports statistically significant gains, compared to both, for all the datasets except AT71. Overall, Flan-XXL and Flan-UL2 outperform other LLMs (Touvron et al., 2023; Jiang et al., 2023) in our ICL experiments and hence reported in Table 3.

**Multilingual Experiments:** Table 4 shows the results for multilingual experiments. On an average ARISE-Iter reports an absolute improvement of 2.1 % points compared to ARISE, the next best model. The results show that our approach is applicable across a typologically diverse set of languages. We find *translation* of synthetically generated English sentences leads to empirically better results as compared to *direct* generation of data in the target language. The results for the former are reported in Table 4. The latter approach results in an absolute drop of 1.27 % points. Further, we also experiment with a setting where we induct rules from dependency parses of all the translations of an input. Here, we observe a performance drop for all the languages, except Hindi. On average there is 0.76 % drop for ARISE-Iter compared to the default setting as reported in Table 4. For Hindi, it reported 78.62 % as compared to 77.41 % in the default setting.

## 5 Conclusion

We propose ARISE, a framework that combines contrastive representation learning, automatic rule induction, data augmentation, IRDF and joint learning via PWS. While PWS is typically employed as a weak supervision approach for labeling unlabeled data, we employ it for verifying synthetically generated labeled documents. Further, we find incorporating syntactic information, instead of strings, via rules leads to gains. Overall, ARISEoutperforms strong competitive baselines under comparable conditions. We also show the effectiveness of combining diverse learning components that enable incorporating complementary information from the limited gold data to achieve state-of-the-art results.

## 6 Limitations

A major challenge with ARISE, currently is the overall training time required to setup a final classifier. We currently use syntactic-ngrams with upto 3 nodes as our features. The search space exponentially increases as the size of the nodes of subtrees further increases, limiting our ability to induce higher-order tree structures as rules. While we currently rely on labeled synthetically generated data, a strength of weak supervision is to incorporate unlabeled data by labeling them. Several real world scenarios often come up where unlabeled data is readily available. It needs to be further investigated whether the synthetically generated labeled data can match the quality of real-world unlabeled data in the context of weak supervision. The current work does not explore this line of work, though it seems to be an important question to be addressed.

## 7 Ethics Statement

All experiments conducted in this study utilize only publicly available datasets. We used publicly hosted APIs of GPT and Claude for synthetic data generation. The prompts included guardrails in the form of instructions to avoid generating problematic content.

## References


Divya Jyoti Bajpai, Ayush Maheshwari, Manjesh Hanawal, and Ganesh Ramakrishnan. 2024. FAIR: Filtering of automatically induced rules. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 573–588, St. Julian's, Malta. Association for Computational Linguistics.

Roy Bar-Haim, Indrajit Bhattacharya, Francesco Dinuzzo, Amrita Saha, and Noam Slonim. 2017. Stance classification of context-dependent claims. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 251–261, Valencia, Spain. Association for Computational Linguistics.

Chris Biemann, Lachezar Krumov, Stefanie Roos, and Karsten Weihe. 2016. Network motifs are a powerful tool for semantic distinction. *Towards a Theoretical Framework for Analyzing Complex Linguistic Networks*, pages 83–105.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

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

Oishik Chatterjee, Ganesh Ramakrishnan, and Sunita Sarawagi. 2020. Robust data programming with precision-guided labeling functions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3397–3404.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR.

Arman Cohan, Waleed Ammar, Madeleine van Zuylen, and Field Cady. 2019. Structural scaffolds for citation intent classification in scientific publications. 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 3586–3596, Minneapolis, Minnesota. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Marie-Catherine de Marneffe, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, and Christopher D. Manning. 2014. Universal Stanford dependencies: A cross-linguistic typology. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 4585–4592, Reykjavik, Iceland. European Language Resources Association (ELRA).

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.

Timothy Dozat and Christopher D Manning. 2016. Deep biaffine attention for neural dependency parsing. *arXiv preprint arXiv:1611.01734*.

John Firth. 1957. A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*, pages 10–32.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, et al. 2023. Massive: A 1m-example multilingual natural language understanding dataset with 51 typologically-diverse languages. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Jay Gala, Pranjal A Chitale, A K Raghavan, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar M, Janki Atul Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, Pratyush Kumar, Mitesh M Khapra, Raj Dabre, and Anoop Kunchukuttan. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *Transactions on Machine Learning Research*.

Boris Galitsky and Dmitry I Ilvovsky. 2019. Least general generalization of the linguistic structures. In *FCA4AI@ IJCAI*, pages 39–44.

Yoav Goldberg and Jon Orwant. 2013. A dataset of syntactic-ngrams over time from a very large corpus of English books. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 241–247, Atlanta, Georgia, USA. Association for Computational Linguistics.

Shai Gretz, Roni Friedman, Edo Cohen-Karlik, Assaf Toledo, Dan Lahav, Ranit Aharonov, and Noam Slonim. 2020. A large-scale dataset for argument quality ranking: Construction and analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7805–7813.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Yiping Jin, Dittaya Wanvarie, and Phu T. V. Le. 2022. Learning from noisy out-of-domain corpus using dataless classification. *Natural Language Engineering*, 28(1):39–69.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020a. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 18661–18673. Curran Associates, Inc.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020b. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673.

Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213.

Terry Koo and Michael Collins. 2010. Efficient third-order dependency parsers. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1–11, Uppsala, Sweden. Association for Computational Linguistics.

Suraj Kothawade, Vishal Kaushal, Ganesh Ramakrishnan, Jeff Bilmes, and Rishabh Iyer. 2022. Prism: A rich class of parameterized submodular information measures for guided data subset selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10238–10246.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. 2019. An evaluation dataset for intent classification and out-of-scope prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1311–1316, Hong Kong, China. Association for Computational Linguistics.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Yen-Ting Lin, Alexandros Papangelis, Seokhwan Kim, Sungjin Lee, Devamanyu Hazarika, Mahdi Namazifar, Di Jin, Yang Liu, and Dilek Hakkani-Tur. 2023. Selective in-context data augmentation for intent detection using pointwise V-information. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1463–1476, Dubrovnik, Croatia. Association for Computational Linguistics.

Fangyu Liu, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2021. Fast, effective, and self-supervised: Transforming masked language models into universal lexical and sentence encoders. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1442–1459, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.

Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. 2019a. Benchmarking natural language understanding services for building conversational agents. *arXiv preprint arXiv:1903.05566*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ayush Maheshwari, Oishik Chatterjee, Krishnateja Killamsetty, Ganesh Ramakrishnan, and Rishabh Iyer. 2021. Semi-supervised data programming with subset selection. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4640–4651.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.

Mohsen Mesgar, Thy Thy Tran, Goran Glavaš, and Iryna Gurevych. 2023. The devil is in the details: On models and training regimes for few-shot intent classification. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1846–1857, Dubrovnik, Croatia. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

GD Plotkin. 1970. A note on inductive generalization. machine intelligence, 5: 153-163, 1970.

Gordon D Plotkin. 1971. A further note on inductive generalization, machine intelligence 6. *Elsevier North-Holland, New York*, 101:124.

RJ Popplestone. 1970. An experiment in automatic induction. *Machine Intelligence*, 5:203–215.

Reid Pryzant, Ziyi Yang, Yichong Xu, Chenguang Zhu, and Michael Zeng. 2022. Automatic rule induction for efficient semi-supervised learning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 28–44, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Luc De Raedt. 2010. *Logic of Generality*, pages 624–631. Springer US, Boston, MA.

Alexander Ratner, Stephen H. Bach, Henry R. Ehrenberg, Jason Alan Fries, Sen Wu, and Christopher Ré. 2017. Snorkel: Rapid training data creation with weak supervision. *CoRR*, abs/1711.10160.

Mohammad Raza, Sumit Gulwani, and Natasa Milic-Frayling. 2014. Programming by example using least general generalizations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 28(1).

Bhavuk Singhal, Ashim Gupta, V P Shivasankaran, and Amrith Krishna. 2023. IntenDD: A unified contrastive learning approach for intent detection and discovery. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14204–14216, Singapore. Association for Computational Linguistics.

Ryan Smith, Jason A. Fries, Braden Hancock, and Stephen H. Bach. 2024. Language models in the loop: Incorporating prompting into weak supervision. *ACM / IMS J. Data Sci.*, 1(2).

Yi Tay, Mostafa Dehghani, Vinh Q Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Dara Bahri, Tal Schuster, Steven Zheng, et al. 2022. Ul2: Unifying language learning paradigms. In *The Eleventh International Conference on Learning Representations*.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Shantanu Thakoor, Simoni Shah, Ganesh Ramakrishnan, and Amitabha Sanyal. 2018. Synthesis of programs from multimodal datasets. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Peter D Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37:141–188.

Paroma Varma and Christopher Ré. 2018. Snuba: Automating weak supervision to label training data. In *Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases*, volume 12, page 223. NIH Public Access.

Ivan Vulić, Iñigo Casanueva, Georgios Spithourakis, Avishek Mondal, Tsung-Hsien Wen, and Paweł Budzianowski. 2022. Multi-label intent detection via contrastive task specialization of sentence encoders. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7544–7559, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ivan Vulić, Pei-Hao Su, Samuel Coope, Daniela Gerz, Paweł Budzianowski, Iñigo Casanueva, Nikola Mrkšić, and Tsung-Hsien Wen. 2021. ConvFiT: Conversational fine-tuning of pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1151–1168, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Zhenyu Wu, Yaoxiang Wang, Jiacheng Ye, Zhiyong Wu, Jiangtao Feng, Jingjing Xu, and Yu Qiao. 2023. OpenICL: An open-source framework for in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 489–498, Toronto, Canada. Association for Computational Linguistics.

Zhuofeng Wu, Sinong Wang, Jiatao Gu, Madian Khabsa, Fei Sun, and Hao Ma. 2020. Clear: Contrastive learning for sentence representation. *arXiv preprint arXiv:2012.15466*.

David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196, Cambridge, Massachusetts, USA. Association for Computational Linguistics.

Asaf Yehudai and Elron Bendel. 2024. When llms are unfit use fastfit: Fast and effective text classification with many classes. *Preprint*, arXiv:2404.12365.

Daniel Zeman, Joakim Nivre, Mitchell Abrams, Elia Ackermann, Noëmi Aepli, Hamid Aghaei, Željko Agić, Amir Ahmadi, Lars Ahrenberg, Chika Kennedy Ajede, Salih Furkan Akkurt, Gabrielė Aleksandravičiūtė, Ika Alfina, Avner Algom, Khalid Alnajjar, Chiara Alzetta, Erik Andersen, Lene Antonsen, Tatsuya Aoyama, Katya Aplonova, Angelina Aquino, Carolina Aragon, Glyd Aranes, Maria Jesus Aranzabe, Bilge Nas Arıcan, Hόrunn Arnardóttir, Gashaw Arutie, Jessica Naraiswari Arwidarasti, Masayuki Asahara, Katla Ásgeirsdóttir, Deniz Baran Aslan, Cengiz Asmazoğlu, Luma Ateyah, Furkan Atmaca, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, Mariana Avelãs, Elena Badmaeva, Keerthana Balasubramani, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, Verginica Barbu Mititelu, Starkaður Barkarson, Rodolfo Basile, Victoria Basmov, Colin Batchelor, John Bauer, Seyyit Talha Bedir, Shabnam Behzad, Kepa Bengoetxea, İbrahim Benli, Yifat Ben Moshe, Gözde Berk, Riyaz Ahmad Bhat, Erica Biagetti, Eckhard Bick, Agnė Bielinskienė, Kristín Bjarnadóttir, Rogier Blokland, Victoria Bobicev, Loïc Boizou, Emanuel Borges Völker, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Adriane Boyd, Anouck Braggaar, António Branco, Kristina Brokaitė, Aljoscha Burchardt, Marisa Campos, Marie Candito, Bernard Caron, Gauthier Caron, Catarina Carvalheiro, Rita Carvalho, Lauren Cassidy, Maria Clara Castro, Sérgio Castro, Tatiana Cavalcanti, Gülşen Cebiroğlu Eryiğit, Flavio Massimiliano Cecchini, Giuseppe G. A. Celano, Slavomír Čéplö, Neslihan Cesur, Savas Cetin, Özlem Çetinoğlu, Fabricio Chalub, Liyanage Chamila, Shweta Chauhan, Ethan Chi, Taishi Chika, Yongseok Cho, Jinho Choi, Jayeol Chun, Juyeon Chung, Alessandra T. Cignarella, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Daniela Corbetta, Francisco Costa, Marine Courtin, Mihaela Cristescu, Ingerid Løyning Dale, Philemon Daniel, Elizabeth

Davidson, Leonel Figueiredo de Alencar, Mathieu Dehouck, Martina de Laurentiis, Marie-Catherine de Marneffe, Valeria de Paiva, Mehmet Oguz Derin, Elvis de Souza, Arantza Diaz de Ilarraza, Carly Dickerson, Arawinda Dinakaramani, Elisa Di Nuovo, Bamba Dione, Peter Dirix, Kaja Dobrovoljc, Adrian Doyle, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Christian Ebert, Hanne Eckhoff, Masaki Eguchi, Sandra Eiche, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Olga Erina, Tomaž Erjavec, Farah Essaidi, Aline Etienne, Wograine Evelyn, Sidney Facundes, Richárd Farkas, Federica Favero, Jannatul Ferdaousi, Marília Fernanda, Hector Fernandez Alcalde, Amal Fethi, Jennifer Foster, Cláudia Freitas, Kazunori Fujita, Katarína Gajdošová, Daniel Galbraith, Federica Gamba, Marcos Garcia, Moa Gärdenfors, Fabrício Ferraz Gerardi, Kim Gerdes, Luke Gessler, Filip Ginter, Gustavo Godoy, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta González Saavedra, Bernadeta Griciūtė, Matias Grioni, Loïc Grobol, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Tunga Güngör, Nizar Habash, Hinrik Hafsteinsson, Jan Hajič, Jan Hajič jr., Mika Hämäläinen, Linh Hà Mỹ, Na-Rae Han, Muhammad Yudistira Hanifmuti, Takahiro Harada, Sam Hardwick, Kim Harris, Dag Haug, Johannes Heinecke, Oliver Hellwig, Felix Hennig, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Marivel Huerta Mendez, Jena Hwang, Takumi Ikeda, Anton Karl Ingason, Radu Ion, Elena Irimia, Ọlájídé Ishola, Artan Islamaj, Kaoru Ito, Siratun Jannat, Tomáš Jelínek, Apoorva Jha, Katharine Jiang, Anders Johannsen, Hildur Jónsdóttir, Fredrik Jørgensen, Markus Juutinen, Hüner Kaşıkara, Nadezhda Kabaeva, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Neslihan Kara, Ritván Karahóǧa, Andre Kåsen, Tolga Kayadelen, Sarveswaran Kengatharaiyer, Václava Kettnerová, Jesse Kirchner, Elena Klementieva, Elena Klyachko, Arne Köhn, Abdullatif Köksal, Kamil Kopacewicz, Timo Korkiakangas, Mehmet Köse, Alexey Koshevoy, Natalia Kotsyba, Jolanta Kovalevskaitė, Simon Krek, Parameswari Krishnamurthy, Sandra Kübler, Adrian Kuqi, Oğuzhan Kuyrukçu, Aslı Kuzgun, Sookyoung Kwak, Kris Kyle, Veronika Laippala, Lorenzo Lambertino, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Phng Lê Hồng, Alessandro Lenci, Saran Lertpradit, Herman Leung, Maria Levina, Lauren Levine, Cheuk Ying Li, Josie Li, Keying Li, Yixuan Li, Yuan Li, KyungTae Lim, Bruna Lima Padovani, Yi-Ju Jessica Lin, Krister Lindén, Yang Janet Liu, Nikola Ljubešić, Olga Loginova, Stefano Lusito, Andry Luthfi, Mikko Luukko, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Menel Mahamdi, Jean Maillard, Ilya Makarchuk, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Büşra Marşan, Cătălina Mărănduc, David Mareček, Katrin Marheinecke, Stella Markantonatou, Héctor Martínez Alonso, Lorena Martín Rodríguez, André Martins, Cláudia Martins, Jan Mašek, Hiroshi Matsuda, Yuji Matsumoto, Alessandro Mazzei, Ryan McDonald, Sarah McGuinness, Gustavo Mendonça, Tatiana Merzhevich, Niko Miekka, Aaron Miller, Karina Mischenkova, Anna Missilä, Cătălin Mititelu, Maria Mitrofan, Yusuke Miyao, AmirHossein Mojiri Foroushani, Judit Molnár, Amirsaeid Moloodi, Simonetta Montemagni, Amir More, Laura Moreno Romero, Giovanni Moretti, Shinsuke Mori, Tomohiko Morioka, Shigeki Moro, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Robert Munro, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Mariam Nakhlé, Juan Ignacio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, Manuela Nevaci, Lng Nguyễn Thị, Huyền Nguyễn Thị Minh, Yoshihiro Nikaido, Vitaly Nikolaev, Rattima Nitisaroj, Alireza Nourian, Hanna Nurmi, Stina Ojala, Atul Kr. Ojha, Hulda Óladóttir, Adédayọ Olúòkun, Mai Omura, Emeka Onwuegbuzia, Noam Ordan, Petya Osenova, Robert Östling, Lilja Øvrelid, Şaziye Betül Özateş, Merve Özçelik, Arzucan Özgür, Balkız Öztürk Başaran, Teresa Paccosi, Alessio Palmero Aprosio, Anastasia Panova, Hyunji Hayley Park, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Guilherme Paulino-Passos, Giulia Pedonese, Angelika Peljak-Łapińska, Siyao Peng, Siyao Logan Peng, Rita Pereira, Sílvia Pereira, Cenel-Augusto Perez, Natalia Perkova, Guy Perrier, Slav Petrov, Daria Petrova, Andrea Peverelli, Jason Phelan, Jussi Piitulainen, Yuval Pinter, Clara Pinto, Tommi A Pirinen, Emily Pitler, Magdalena Plamada, Barbara Plank, Thierry Poibeau, Larisa Ponomareva, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Robert Pugh, Tiina Puolakainen, Sampo Pyysalo, Peng Qi, Andreia Querido, Andriela Rääbis, Alexandre Rademaker, Mizanur Rahoman, Taraka Rama, Loganathan Ramasamy, Joana Ramos, Fam Rashel, Mohammad Sadegh Rasooli, Vinit Ravishankar, Livy Real, Petru Rebeja, Siva Reddy, Mathilde Regnault, Georg Rehm, Arij Riabi, Ivan Riabov, Michael Rießler, Erika Rimkutė, Larissa Rinaldi, Laura Rituma, Putri Rizqiyah, Luisa Rocha, Eiríkur Rögnvaldsson, Ivan Roksandic, Mykhailo Romanenko, Rudolf Rosa, Valentin Roșca, Davide Rovati, Ben Rozonoyer, Olga Rudina, Jack Rueter, Kristján Rúnarsson, Shoval Sadde, Pegah Safari, Aleksi Sahala, Shadi Saleh, Alessio Salomoni, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Ezgi Sanıyar, Dage Särg, Marta Sartor, Mitsuya Sasaki, Baiba Saulīte, Yanin Sawanakunanon, Shefali Saxena, Kevin Scannell, Salvatore Scarlata, Nathan Schneider, Sebastian Schuster, Lane Schwartz, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Syeda Shahzadi, Mo Shen, Atsuko Shimada, Hiroyuki Shirasu, Yana Shishkina, Muh Shohibussirri, Maria Shvedova, Janine Siewert, Einar Freyr Sigurðsson, João Silva, Aline Silveira, Natalia Silveira, Sara Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Haukur Barri Símonarson, Kiril Simov, Dmitri Sitchinava, Ted Sither, Maria Skachedubova, Aaron Smith, Isabela Soares-Bastos, Per Erik Solberg, Barbara Sonnenhauser, Shafi Sourov, Rachele Sprugnoli, Vivian Stamou, Steinhór Steingrímsson, Antonio Stella, Abishek Stephen, Milan Straka, Emmett Strickland, Jana Strnadová, Alane Suhr, Yogi Lesmana Sulestio, Umut Sulubacak, Shingo Suzuki, Daniel Swanson, Zsolt Szántó, Chihiro Taguchi, Dima Taji, Fabio Tamburini, Mary Ann C. Tan, Takaaki Tanaka, Dipta Tanaya, Mirko Tavoni, Samson Tella, Isabelle Tellier, Marinella Testori, Guillaume Thomas, Sara Tonelli, Liisi Torga, Marsida Toska, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Utku Türk, Francis Tyers, Sveinbjörn Hórðarson, Vilhjálmur Horsteinsson, Sumire Uematsu, Roman Untilov, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Andrius Utka, Elena Vagnoni, Sowmya Vajjala, Socrates Vak, Rob van der Goot, Martine Vanhove, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Uliana Vedenina, Giulia Venturi, Veronika Vincze, Natalia Vlasova, Aya Wakasa, Joel C. Wallenberg, Lars Wallin, Abigail Walsh, Jonathan North Washington, Maximilan Wendt, Paul Widmer, Shira Wigderson, Sri Hartati Wijono, Seyi Williams, Mats Wirén, Christian Wittern, Tsegay Woldemariam, Tak-sum Wong, Alina Wróblewska, Mary Yako, Kayo Yamashita, Naoki Yamazaki, Chunxiao Yan, Koichi Yasuoka, Marat M. Yavrumyan, Arife Betül Yenice, Olcay Taner Yıldız, Zhuoran Yu, Arlisa Yuliawati, Zdeněk Žabokrtský, Shorouq Zahra, Amir Zeldes, He Zhou, Hanzhi Zhu, Yilun Zhu, Anna Zhuravleva, and Rayan Ziane. 2023. Universal dependencies 2.12. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Jianguo Zhang, Trung Bui, Seunghyun Yoon, Xiang Chen, Zhiwei Liu, Congying Xia, Quan Hung Tran, Walter Chang, and Philip Yu. 2021a. Few-shot intent detection via contrastive pre-training and fine-tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1906–1912, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jieyu Zhang, Cheng-Yu Hsieh, Yue Yu, Chao Zhang, and Alexander Ratner. 2022a. A survey on programmatic weak supervision. *Preprint*, arXiv:2202.05433.

Jieyu Zhang, Yue Yu, , Yujing Wang, Yaming Yang, Mao Yang, and Alexander Ratner. 2021b. Wrench: A comprehensive benchmark for weak supervision. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Yuhao Zhang, Hongji Zhu, Yongliang Wang, Nan Xu, Xiaobo Li, and Binqiang Zhao. 2022b. A contrastive framework for learning sentence representations from pairwise and triple-wise perspective in angular space. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4892–4903, Dublin, Ireland. Association for Computational Linguistics.

Dawei Zhu, Xiaoyu Shen, Marius Mosbach, Andreas Stephan, and Dietrich Klakow. 2023. Weaker than you think: A critical look at weakly supervised learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14229–14253, Toronto, Canada. Association for Computational Linguistics.

# A Example Appendix

This is an appendix.