Unsupervised Testing of NLU models with Multiple Views

Radhika Arava, Matthew Trager, Boya Yu, Mohamed Abdelhady
Amazon Alexa AI
{aravar, mttrager, boyayu,mbdeamz}@amazon.com

Abstract

In Natural Language Understanding (NLU) systems in voice assistants, new domains are added on a regular basis. This poses the practical problem of evaluating the performance of NLU models on domains where no manually annotated data is available. In this paper, we present an unsupervised testing method that we call *Cross-View Testing* (CVT) for ranking multiple intent classification models using only unlabeled test data. The approach relies on a number of labeling functions to automatically annotate test data in the target domain. The labeling functions include intent classification models trained on different domains, as well as heuristic rules. Specifically, we combine the annotations of multiple models with different output spaces by training a combiner model on synthetic data. In our experiments, the proposed model outperforms the target models by very large margins, and its predictions can be used as a proxy of ground truth for unsupervised model evaluation.

1 Introduction

In spoken language understanding systems utilized in virtual voice assistants such as Amazon Alexa, Apple Siri and Google Home, third-party domains (skills) are added on a regular basis. An important challenge in domain adaptation is to quickly develop a model for a new domain given limited supervision from the existing domains. Labeled live utterances are generally not available for either training or testing of the NLU models of such domains. A synthetic training data set can be generated by sampling from finite-state transducers (FSTs) that are created using the carrier phrases defined by a domain developer. Depending on the number and linguistic diversity of the carrier phrases, transfer learning techniques can be applied on the resulting training set to train an accurate NLU model. In addition, Semi-Supervised Learning (SSL) approaches can be applied to augment the training set with automatically annotated live utterances (Soto and Arkoudas, 2021). The lack of annotated testing data makes proper model evaluation and comparison extremely challenging.

In this paper, we discuss the problem of *unsupervised model comparison*. Our goal is to compare the performance of two NLU models A and B using only *unlabeled* real data for testing, in addition to synthetic labeled data for training. This task is challenging because both of the models that we wish to compare can generally fit the training data perfectly. In particular, if model A and model B return different labels for the same real utterance u, we do not know a priori which one is correct (or if neither one is).

To solve this problem, we train a model that takes as input the predictions of A and B together with the predictions of different "reference models" $R_1, \ldots, R_n$ on each utterance as extra signals. The model then outputs a predicted intent for the given utterance, which is treated as a pseudo-ground truth. Perhaps surprisingly, the reference models can be trained on *different domains* and do not have the same output spaces compared to A and B. In our setting, the reference models are trained on different skills from the same skill category as the target skill. Although individually none of these reference models can be used to predict the correct label, we show that collectively they can be fused using our model to decide which one among A and B is most likely to be correct. This is possible because there is sufficient overlap in the intent and slot definitions among skills from the same category.

We introduce a weak supervision approach that merges the annotations from multiple reference models. The approach is named Cross-View Testing because each reference model is an out-of-domain expert that provides a different view of an utterance intent. Using this approach, we discuss several methods for combining the predictions of the labeling functions, including a few classical

models that use predictions as categorical input features, and an attention-based approach where the reference models attend to the embedded output of the target models A and B ("attention voting"). We also propose some heuristic strategies that can be incorporated in any of these approaches. In our experiments, all of these methods outperform the target models and can be successfully used for unsupervised testing.

2 Related Work

Our work is related to the general task of *domain adaptation*, where the goal is to transfer knowledge from one or multiple source domains to a different target domain. Previous works on this topic have introduced a distinction between "data-driven" and "model-driven" approaches (Lee and Jha, 2019). Data-driven methods make use of training data from source domains with techniques such as feature augmentation (Daumé, 2007; Kim et al., 2016). Model-driven approaches directly consider the outputs of "expert models" previously trained on different datasets (Kim et al., 2017; Jha et al., 2018; Kim et al., 2018; Guo et al., 2018). The method we adopt here is model-based, and is most similar to the "bag of experts" strategy from (Jha et al., 2018). However, the typical motivation for domain adaptation is to reduce the amount of task-specific data for training, while we address the lack of data for *testing* in the target domain. Our work is also related to the problem of *aggregating* the predictions of multiple models without supervision. This task has been studied for applications in *crowdsourcing* to merge inconsistent annotations (Karger et al., 2013; Raykar et al., 2010) and also in the context of weakly supervised learning (Lison et al., 2020).

3 Cross-View Testing (CVT)

We introduce a weak supervision approach that merges the annotations from multiple labeling functions. Labeling function are based on models that take as input an utterance and return its predicted intent. Each model is pre-trained to recognize only the subset of possible intents supported by either a first-party or a third-party domain of a commercial voice assistant. The approach is named Cross-View Testing because each model is an out-of-domain expert that provides a different view of an utterance intent.

In the following we write $\mathcal{X}$ for the instance space of all possible utterances and $\mathcal{Y}^{(d)}$ for the space of possible intent labels for the skill/domain d. For each skill d we aim to evaluate and compare the performance of two *target* models $A^{(d)}$ and $B^{(d)}$. However, we assume that we do not have access to any real labeled data for testing. Instead, we use synthetically generated labeled data for training and real but unlabeled utterances for testing. We also make use of a set of trained *reference* models $R_1, \ldots, R_n$, where each model $R_i : \mathcal{X} \rightarrow \mathcal{Y}^{(d_i)}$ can have a different output space.

Target and reference models. The methods discussed in this paper are generic and can in principle be applied for any set of pre-trained target and reference models. For all skills in our experimental setup, model $A^{(d)}$ is a n-gram based multinomial logistic regression model and model $B^{(d)}$ is a BiLSTM-based deep neural net with pre-trained subword embeddings. The reference models we used have the same two architectures as the target models but were trained on different skills/domains from the same category.

4 Combining Labeling Functions

In this section, we discuss different strategies for combining the labels of target models and reference models into a single output intent. We will later apply these approaches on unlabeled real data to obtain pseudo-ground truth labels for testing. All of the methods we discuss are trained using only synthetically generated data. We remark that although these methods generally outperform the two target models, they cannot be used at inference time due to the high latency caused by evaluating multiple models on the same utterance.

4.1 Classical Approaches

We first experiment with a few classical machine learning methods. We consider the predictions of all reference models on a given utterance u as categorical features for predicting the correct intent for u. After applying a one-hot encoding to these features, we train several models using synthetic data for the target skill. We present results for the following three approaches:

- **Ridge**: a linear ridge regression model where the regularization parameter is selected using cross-validation.
- **MLP**: a multi-layer perceptron with ReLU activations, using three hidden layers with sizes

100, 50, 25.

- **Stack**: a stacked classification model, where a logistic regression layer is applied to the output of two models: an MLP classifier as described above, and a ridge regression classifier where the inputs are indicator features for all pairs of reference model intent predictions (*i.e.*, the quadratic interaction features from the one-hot encoding). This is analogous the approach discussed in Cheng et al. (2016).

4.2 Attention Voting

In this approach, we use the predictions of the reference models to "vote" for the predictions of the two target models, based on dot-product attention. More precisely, we have embedded representations of reference intents $\mathbf{R} \in \mathbb{R}^{n \times d}$ and target intents $\mathbf{T} \in \mathbb{R}^{2 \times d}$, where n is the number of reference models and d is the embedding dimension. We then consider the following attention weights

$$\mathbf{A} = \mathrm{softmax}_{full}\left(\frac{\mathbf{R}\mathbf{T}^{\top}}{\sqrt{d}}\right) \in \mathbb{R}^{n \times 2}.$$

Here, unlike the typical formulation of the attention mechanism, $\mathrm{softmax}_{full}$ applies softmax over the *entire* (flattened) attention weight matrix. The reason for this normalization choice is that we do not want irrelevant reference intents to have high attention scores. As shown in Figure 1, the final intent encoding is then given by $\mathbf{v} = \mathrm{concat}(\mathbf{AT}) \in \mathbb{R}^{nd}$ where concat denotes concatenation of the rows. Each row $\mathbf{AT}$ is a (scaled) convex combination of the embedded intents for the target models A and B. In particular, if the target models predict the same intent, then the rows of $\mathbf{AT}$ are all scalar multiples of each other. We use the intent encoding vector $\mathbf{v}$ as input to an MLP layer for multi-class classification. Note that the final layer can in principle predict any possible intent for the target skill, so the output of our model does not necessarily coincide with the prediction of either of the target models.

5 Additional Labeling Functions

In the previous section, we presented approaches for merging the annotations of the different pretrained intent classification models. In this section, we discuss some heuristic rules that can be used as additional labeling functions for CVT.

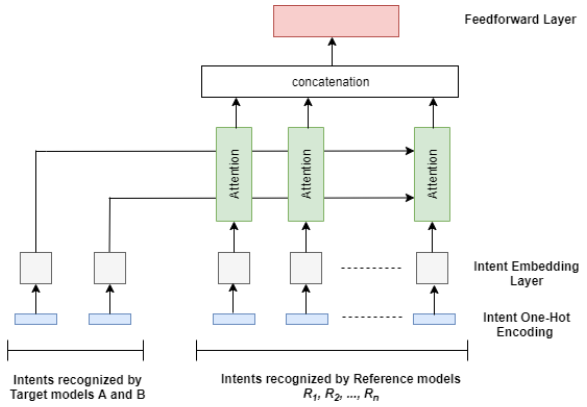

Figure 1: Network architecture of Attention Voting combination method.

5.1 Slot Intent Lexical Matching (SILM)

We have observed in our data analysis that skill developers usually choose named entity tags that are related lexically or at least semantically to the intent of the given utterance. Based on this observation, we extracted a Slot Intent Lexical Match (SILM) signal such that if say target model A's intent lexically matches with any of the entity tags predicted by any of the two target models, then the SILM signal suggests that model A possibly has the right output. If exactly one of the two target models match the SILM signal, we override the combiner model output with the SILM prediction.

5.2 Fallback Intent Substitution (FS)

Some of the skills that we consider include a special intent to handle random and out-of-domain utterances called "Fallback Intent" (FBI). This intent is meant to be a label for utterances that were incorrectly assigned to the skill. We have observed that both target models sometimes over-predict FBI, due to the relatively large presence of FBI in our synthetic training sets. For this reason, we introduce a simple practical rule that addresses this issue: when exactly one of the two target models predicts FBI (say $A^{(d)}$), we select the intent predicted by the other model ($B^{(d)}$) as proxy for ground truth, ignoring the output of the combiner model. If neither or even both models predict FBI, we use the output of the combiner model.

6 Experiments

We first report results for intrinsic performance evaluation of the CVT labeling functions on the intent classification task, in terms of (relative average) intent classification error rate (ICER). We then

Table 1: Intent Classification results for two target models (A and B) and the different Cross-View approaches. Percentages indicate relative average ICER improvement compared to the average ICER of models A and B (averaged over all skills in the category). We also indicate in parenthesis the number of skills with statistically significant improvements over the baseline (out of 12), for a 95% normal-based confidence interval.

Model	Games	Music
A	6.9%	-11.8%
B	-6.9%	11.8%
Ridge	0.8% (6)	12.3% (5)
Ridge+FS	19.7% (8)	25.5% (7)
Ridge+SILM	13.1% (8)	28.7% (8)
MLP	1.0% (8)	36.1% (7)
MLP+FS	17.1% (8)	49.2% (8)
MLP+SILM	13.2% (8)	50.7% (8)
Stack	3.9% (7)	38.4% (6)
Stack+FS	19.9% (9)	50.6% (8)
Stack+SILM	13.1% (8)	49.4% (8)
AV	7.1% (8)	22.4% (4)
AV+FS	19.6% (9)	47.2% (7)
AV+SILM	13.5% (8)	49.1% (7)

report the extrinsic performance evaluation results of the proposed approach into the downstream task of automatic model evaluation and ranking.

For both types of evaluation, we noted that jointly applying FS and SILM never significantly improved over applying the best of these two heuristic rules. For compactness, we thus only report results including either FS or SILM.

6.1 Data

In order to evaluate performance, we apply our proposed CVT approaches on 12 skills in the Games category and 12 skills in the Music category. For these skills, we have access to a set of labeled real utterances that we only use to evaluate the performance of our methods. For each skill, we use models trained on all other skills of the same category as weak labels.

6.2 Intent classification results

Without ground truth annotations, we do not know a priori which of the two target models has better performance. For this reason, we use the average ICER of the two target models as a baseline for the intent classification task. This average can be seen as the expected performance when selecting the prediction of one of the two target models randomly with uniform probability. Table 1 presents the average ICER of different approaches, relative to this average baseline. More precisely, the percentage for each model M indicates $\frac{\text{avgICER}(M)}{\text{avgICER}(AB)} - 1$ where avgICER is the average ICER over all skills in the category, and AB indicates the baseline average (so $\text{avgICER}(AB) = \frac{1}{2}(\text{avgICER}(A) + \text{avgICER}(B))$. All of our approaches outperform the baseline in both categories and generally by very large margins (particularly using FS and SILM). In fact, our methods perform on average better than the best of the two models, despite using only weak additional knowledge as input (note model A is better than model B for Games category, while the opposite is true for the Music category). We also indicate in parenthesis the number of skills with statistically significant improvements over the baseline (for a 95% normal-based confidence interval).

6.3 Unsupervised testing results

We next evaluate whether the proposed methods are able to compare the performance of the two target models (A and B) on a given skill without using manually-labeled test utterances. In Figure 2, we show the difference between the true ICER of model A and model B with the difference of the ICER of the same models, estimated using pseudo-ground truth labels from the CVT model. For compactness we only include plots for approaches including FS and SILM. The results show that all methods are able to estimate the difference quite accurately, and the Pearson correlation between the estimated and true difference (shown on top of each plot) is often above .9. We note, however, that for one skill in the Games category the basic assumption for the SILM signal is not satisfied, and this causes the mean ICER estimates to be off and the Pearson correlation to be low (second row of Figure 2). In Table 2, we count the number of skills for which the CVT approach ranks model A and model B correctly. This corresponds to the number of dots in the green quadrants in each of the plots in Figure 2. Although we do not achieve perfect accuracy, we see from Figure 2 that the ranking is inconsistent mostly for skills where (true and estimated) ICER of the two target models is very similar.

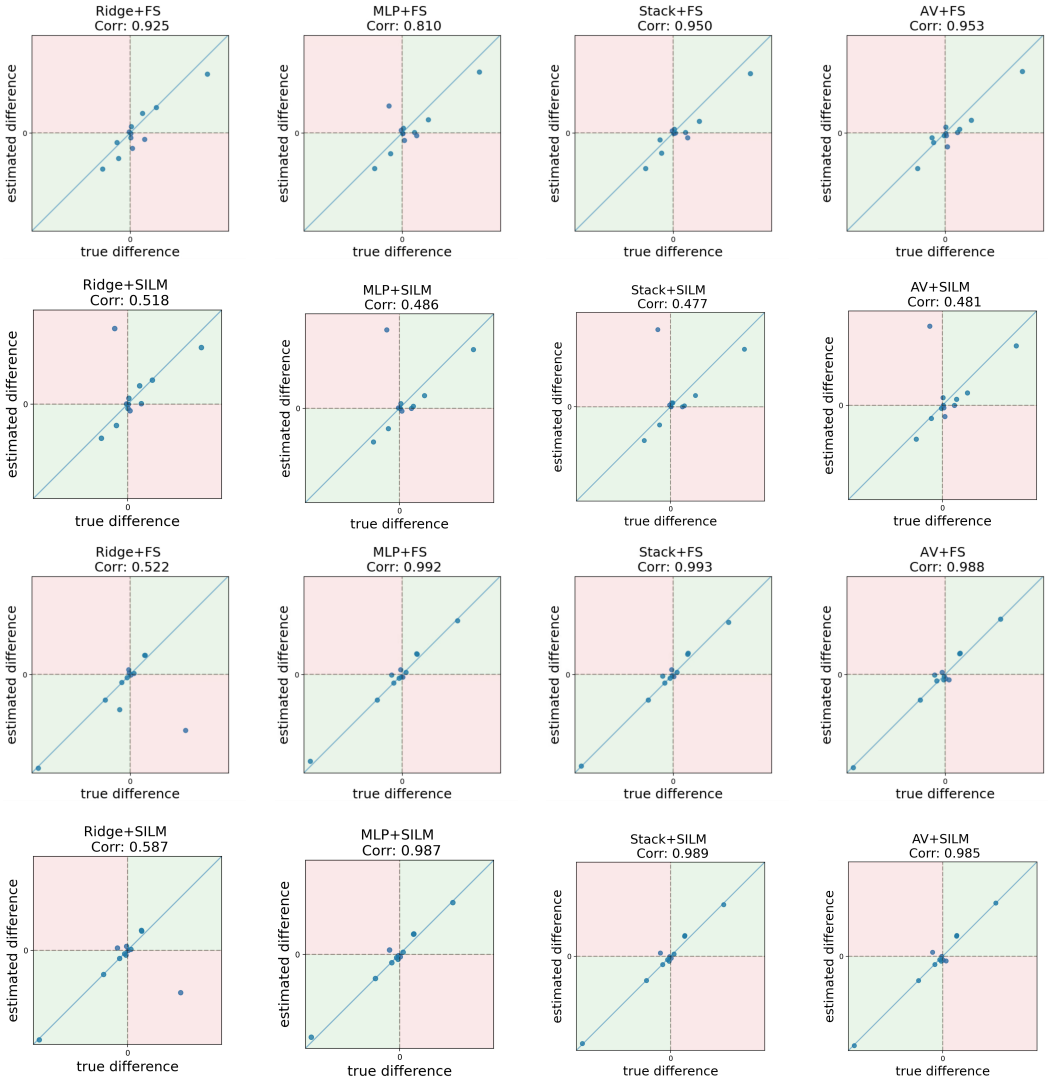

Figure 2: Comparison between the true difference and the estimated difference in the ICER performance of the target models A and B, for all skills in the Games (top two rows) and Music (bottom two rows) category. Each point corresponds to a skill. Points that lie in the green quadrants are skills where the rankings provided by unsupervised testing and supervised testing agree. The figures show that almost all points are either inside or very close to the green quadrants and incorrectly ranked skills are generally points very close to the origin.

7 Conclusions

We presented several approaches for the unsupervised testing of multiple NLU models without the need of labeled test data. More precisely, we have shown that pre-trained models of different skills/domains can be used to test and rank multiple NLU models in order to decide which of them is more accurate for the intent classification task of a given domain. Using only synthetic data for training, we obtained models that achieve significantly better results than both the original target models.

The general framework that we presented is flexible and allows for many variations. The embeddings that we used for attention voting were trained from scratch but they could also be pre-trained to better encode intent or slot semantic information. In addition to this, it is reasonable to provide the actual utterance tokens as input to the target-reference models. In our experiments, this did not seem to improve the performance of the joint model, but more fine-tuning and experimentation might lead to better results. We also plan to use the learned attention weights to find clusters of similar intents and skills.

Finally, we believe that the fact that reference models improve performance of the target models is intriguing, and we would like to investigate this phenomenon more closely. We hypothesize that in addition to certain skills providing useful cross-view information, there may also be an implicit

Table 2: IC Model Ranking Accuracy (A vs. B) for different CVT approaches

Model	Games	Music
Ridge	6 / 12	6 / 12
Ridge+FS	7 / 12	8 / 12
Ridge+SILM	8 / 12	8 / 12
MLP	6 / 12	8 / 12
MLP+FS	6 / 12	10 / 12
MLP+SILM	8 / 12	10 / 12
Stack	8 / 12	8 / 12
Stack+FS	9 / 12	10 / 12
Stack+SILM	10 / 12	10 / 12
AV	7 / 12	6 / 12
AV+FS	9 / 12	9 / 12
AV+SILM	9 / 12	8 / 12

regularization phenomenon at play. We hope to study this effect by experimenting with more skills and datasets.

References

Heng-Tze Cheng, L. Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, H. Aradhye, Glen Anderson, G. Corrado, Wei Chai, M. Ispir, Rohan Anil, Zakaria Haque, L. Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. 2016. Wide & deep learning for recommender systems. *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*.

Hal Daumé. 2007. Frustratingly easy domain adaptation. *ArXiv*, abs/0907.1815.

Jiang Guo, Darsh J Shah, and Regina Barzilay. 2018. Multi-source domain adaptation with mixture of experts. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4694–4703. Association for Computational Linguistics.

Rahul Jha, Alex Marin, S. Shivaprasad, and I. Zitouni. 2018. Bag of experts architectures for model reuse in conversational language understanding. In *NAACL*.

D. Karger, Sewoong Oh, and D. Shah. 2013. Efficient crowdsourcing for multi-class labeling. In *SIGMETRICS '13*.

Young-Bum Kim, Dongchan Kim, Anjishnu Kumar, and Ruhi Sarikaya. 2018. Efficient large-scale domain classification with personalized attention. Association for Computational Linguistics.

Young-Bum Kim, K. Stratos, and Dongchan Kim. 2017. Domain attention with an ensemble of experts. In *ACL*.

Young-Bum Kim, K. Stratos, and R. Sarikaya. 2016. Frustratingly easy neural domain adaptation. In *COLING*.

Sungjin Lee and Rahul Jha. 2019. Zero-shot adaptive transfer for conversational language understanding. In *AAAI*.

Pierre Lison, A. Hubin, Jeremy Barnes, and Samia Touileb. 2020. Named entity recognition without labelled data: A weak supervision approach. In *ACL*.

V. Raykar, Shipeng Yu, Linda H. Zhao, G. Hermosillo, Charles Florin, L. Bogoni, and L. Moy. 2010. Learning from crowds. *J. Mach. Learn. Res.*, 11:1297–1322.

Victor Soto and Konstantine Arkoudas. 2021. Combining weakly supervised ml techniques for low-resource nlu. In *NAACL HLT*.