Redwood: Using Collision Detection to Grow a Large-Scale Intent Classification Dataset

Stefan Larson and **Kevin Leach**
Vanderbilt University
{firstname.lastname}@vanderbilt.edu

Abstract

Dialog systems must be capable of incorporating new skills via updates over time in order to reflect new use cases or deployment scenarios. Similarly, developers of such ML-driven systems need to be able to add new training data to an already-existing dataset to support these new skills. In intent classification systems, problems can arise if training data for a new skill's intent overlaps semantically with an already-existing intent. We call such cases *collisions*. This paper introduces the task of intent collision detection between multiple datasets for the purposes of growing a system's skillset. We introduce several methods for detecting collisions, and evaluate our methods on real datasets that exhibit collisions. To highlight the need for intent collision detection, we show that model performance suffers if new data is added in such a way that does not arbitrate colliding intents. Finally, we use collision detection to construct and benchmark a new dataset, *Redwood*, which is composed of 451 intent categories from 13 original intent classification datasets, making it the largest publicly available intent classification benchmark.

1 Introduction

As task-oriented dialog systems like Alexa and Siri have become more and more pervasive, tools enabling developers to build custom dialog systems have followed suit. Such tools—like Microsoft's Luis[1], Twilio's Autopilot[2], Rasa[3], and Google's DialogFlow[4]—enable engineers and dialog designers to craft dialog systems composed of *intents*, or core categories of competencies or skills in which the system is knowledgeable and to which the system can respond intelligently. New intents may be added periodically to the dialog system as part of its development and maintenance cycle, or dialog system models may be combined together (e.g., Clarke et al. (2022)).

These phenomena may occur especially in real-world deployments, where datasets for dialog models may be developed, grown, and modified by large (and even disparate) teams over the span of a project's lifetime. Furthermore, dialog system models and their corresponding training datasets are sometimes offered as-a-service or "off-the-shelf" to dialog system builders who might not be fully familiar with the breadth or scope of the pre-existing dataset or model. If the builder adds a new intent to the dataset that overlaps with an existing intent, then the re-trained model's performance can suffer. As such, there is a need for tools and algorithms to help detect when a new intent overlaps—that is, *collides*—with an already-existing intent category.

In this paper, we introduce the challenge of *intent collision detection*, and develop several algorithms for determining whether a candidate intent category collides with another intent category. To do so, we curate and release a meta-dataset of 722 intents from 13 existing datasets. This graph-like meta-dataset consists of annotations indicating tuples of colliding intent pairs (examples of colliding intents can be seen in Table 1). We then introduce several collision detection algorithms and evaluate them on this meta-dataset.

We also use intent collision detection to build *Redwood*, a new intent classification dataset of 451 intent categories. *Redwood* is built by combining 13 smaller datasets. As a comparison, we also build *Redwood-naïve*, which is constructed by naïvely joining together all 13 datasets without arbitrating colliding intents. We find that classifier performance on *Redwood-naïve* to be substantially worse than *Redwood*, showcasing the negative effect of not addressing intent collisions in data.

Upon official release, *Redwood* will by far be the largest openly available intent classification dataset in terms of breadth of intent categories. Our hope is that the new *Redwood* dataset serves as a showcase for intent collision detec-

---

[1] luis.ai   [2] twilio.com/autopilot   [3] rasa.com
[4] google.com/dialogflow

Dataset	Samples		
Snips	*how cold is it in princeton junction*	*will it be chilly in fiji at ten pm*	*is it foggy in shelter island*
Clinc-150	*give me the 7 day forecast*	*what's the temperature like in tampa*	*will it rain today*
MTOP	*what is the weather in new york today*	*how much is it going to rain tomorrow*	*give me the weather for march 13th*
Slurp	*set alarm tomorrow at 6 am*	*make an alarm for 4pm*	*set a wake up call for 10 am*
MTOP	*can you set a warning alarm for 7pm*	*set an alarm for monday at 5pm*	*make an alarm for the 5th*
Clinc-150	*wake me up at noon tomorrow*	*set my alarm for getting up*	*i need you to set alarm for me*
HWU	*how much is 1gbp in usd*	*what's the exchange rates*	*how much is $50 in pounds*
Clinc-150	*tell me five dollars in yen and rubles*	*how many pesos in one dollar us*	*usd to yen is what right now*
Banking-77	*do you know the rate of exchange*	*how is the exchange rate doing*	*what are the current exchange rates*
Clinc-150	*please start calling me mandy*	*I want you to call me this new name*	*the name you should call me is janet*
ACID	*how do i change my name*	*need my name to be updated*	*I need to fix my name in your system*
Banking-77	*where can I find how to change my name*	*details need to be modified*	*after I got married I need to change my name*
Snips	*play magic sam from the thirties*	*play music by blowfly from the seventies*	*play jeff pilson on youtube*
DSTC-8	*I want to hear the song high*	*I would like to listen to touch it on tv*	*I'd like to listen to the way I talk*
HWU	*please play yesterday from beattles*	*I'd like to hear queen's barcelona*	*play daft punk*
MetalWOz	*help me find restaurants in miami fl*	*I need help finding a place to eat*	*I need to find an italian restaurant in denver*
DSTC-8	*can you help find a place to eat*	*I'm looking for a filipino place to eat*	*I want to find a restaurant in albany*
HWU	*find me a nice restaurant for dinner*	*where can I get shawarma in this area*	*what's the best chicken place near me*
Outlier	*what is my balance*	*update me on my account balance*	*let me know how much money I have*
Clinc-150	*what's my current checking balance*	*what is the total of my bank accounts*	*how much total cash do I have in the bank*
DSTC-8	*I want to know my checking account balance*	*I'd like to check my balance*	*man how much money do I have in the bank*

Table 1: Examples of data that will trigger collisions. Each row of the table displays three samples from a single intent in a particular dataset. Among these three samples, each line collides with an intent category from the other two datasets.

tion as well as a new, publicly-available, large-scale challenge dataset for intent classification models for dialog systems. Both the collision meta-dataset and *Redwood* are publicly available at `github.com/gxlarson/redwood`.

2 Related Work

The Collision Detection Task. We discuss three areas of related work related to our proposed intent collision detection task: generalized zero-shot learning, open set classification, and out-of-domain (or out-of-scope) sample identification.

In generalized zero-shot learning (e.g., Zhang et al. (2022)), a model is trained with data from a set of "seen" label classes (e.g., intents) and, during inference, must identify test samples as belonging to either a "seen" label class or an "unseen" class for which the model has limited auxiliary knowledge (e.g., descriptions of unseen classes, but no concrete training examples).

Both open set classification and out-of-domain sample identification refer to the modeling task of classifying inference samples among label classes seen during training or to identify if the sample belongs to an unknown or undefined label class (e.g., Larson et al. (2019b); Zhang et al. (2021)). Slot-filling models that are trained on B/I/O tags naturally predict the unknown class label as O tags, but for intent classifiers the task is much more challenging since it requires curating viable training data for

an out-of-domain category (i.e., it is challenging to know in advance what types of out-of-domain inputs a system might encounter).

Our proposed task of intent collision detection differs from the aforementioned tasks because "inference" samples need not be considered one at a time, but can instead be grouped together into entire candidate intent categories. This enables considering entirely different modeling tasks like those discussed in Section 3.3. Nevertheless, both our meta-dataset of intent collisions and *Redwood* allow for the evaluation of both zero-shot and generalized zero-shot learning models, and the *Redwood* intent classification dataset includes a substantial number of out-of-domain samples for evaluating open set classification and out-of-domain sample detection.

Intent Classification Corpora. There are several smaller corpora for evaluating intent classification models, some spanning broad domains (e.g., Liu et al. (2019), Larson et al. (2019b), Li et al. (2021)) and others focusing fine-grained evaluation of individual domains (e.g., the Banking-77 corpus (Casanueva et al., 2020) with respect to the personal banking domain). While most datasets are constructed via crowdsourcing, our new Redwood dataset is constructed from both (1) already existing datasets and (2) newly crowdsourced intents.

Dataset Derivation and Combination. Datasets are sometimes formed from other datasets, either
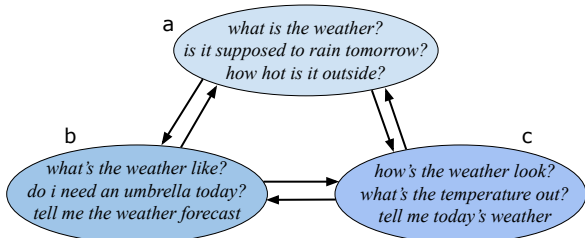
Figure 1: Transitive collisions.

Figure 2: Non-transitive collisions.

by deriving a new dataset from an existing one, or by combining datasets together. The former category include translations of dialog datasets (e.g., (Upadhyay et al., 2018; Xu et al., 2020)) as well as re-formulations of existing datasets into new tasks (e.g., converting a semantic role labeling (SRL) dataset to open information extraction (OIE) data as done in Solawetz and Larson (2021)).

Dataset combination has been used in other fields beyond dialog systems and conversational AI. For instance, Song et al. (2020) combined several speech recognition datasets together to form their *SpeechStew* dataset. As there are no target labels analogous to intents in automatic speech recognition, the creators of *SpeechStew* did not have to consider collisions among intent categories. In this paper, our focus is primarily on dataset combination, but we also derive intent classification data from several turn-based dialog corpora (*MetalWOz* and *DSTC-8*, discussed in Section 3.4).

3 Detecting Collisions

In this section we discuss our proposed challenge, intent collision detection. We begin with a motivating example showing why detecting collisions is important, as well as a formal problem statement. Then, we introduce and evaluate several collision detection baselines on our meta-dataset.

3.1 Motivating Example

As a motivating example, suppose our intent classification system has been trained on the *Clinc-150* dataset (Larson et al., 2019b), an intent classification dataset consisting of 150 intents.[5] The *Clinc-150* dataset includes an intent called weather, which is meant to handle weather-related queries such as "*what's the weather like today*" and "*tell me the weather in New York.*" Suppose further that a new developer or a new team attempts to update the intent classifier with new data that contains a
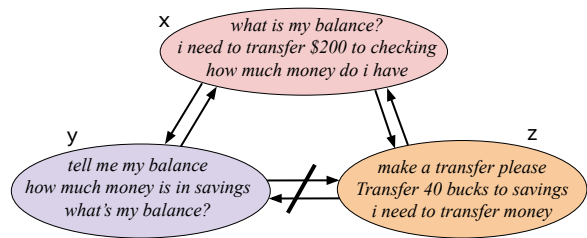
new intent category, such as the get_weather intent from the *HWU* dataset[6] (Liu et al., 2019). In such a scenario, there are now training data samples that overlap substantially, but that are labeled with different intents (weather vs. get_weather in this example). Thus, upon updating the model by training on *HWU*'s get_weather data, the predictive performance on any weather-related inference queries might be split between these two intents. This disparity can also cause unintended consequences downstream in production models, such as calls to database systems that are triggered based on the user's intent.

Indeed, when we train a BERT classifier on the original *Clinc-150* training set, the accuracy on the weather test set is 100%. When we add a *HWU*'s get_weather intent to *Clinc-150* to create a new 151st intent and re-train the BERT classifier, we observe an accuracy score of 60% on the weather test set. This performance drop is a symptom of having added an intent category that collides with another intent category. Such a model—which was trained on colliding intents—could cause unexpected behavior on downstream events, especially if the weather and get_weather intents trigger different business logic workflows or system responses. We note that, while in this example, the colliding weather and get_weather intent names are quite similar, other colliding pairs like Snips' search_screening_event and MetalWOz's movie_listings do not have lexically similar intent names, precluding straightforward string matching of intent names.

3.2 Problem Statement

In this subsection, we formally define our collision detection problem. We first consider a scenario in which we have two intent classification datasets, $\mathcal{A}$ and $\mathcal{B}$, where $A_i \in \mathcal{A}$ and $B_j \in \mathcal{B}$ refer to specific intent categories in each. We say that intent categories A_i and B_j *collide* if there exist a suf-

---

[5] In this paper, dataset names are in *italics* and intent names are in teletype font. Example queries are in *italics* and in quotes if they appear in-line.

[6] Recall from Section 1 that such updates from new teams or new developers may be from routine perfective maintenance during a model's lifetime.

ficient number of queries in A_i that *semantically overlap* with a sufficient number of queries in B_j. This *semantic overlap* can occur when a developer attempts to add new intent categories to a starting training dataset—when an intent classification model trained on the combined dataset $\mathcal{A} \cup \mathcal{B}$ will cause queries belonging to A_i to be classified in B_j (and vice versa).

As an example, suppose we have an intent classifier built from a starting dataset such as *Clinc-150*, which, among other things, contains a `weather` intent category for weather-related inquiries (cf. Section 3.1). Suppose further that we seek to grow this starting dataset by adding datapoints from a candidate dataset such as *HWU* (see Section 3.1, which contains a `get_weather` intent category). If we naïvely combine these two datasets together, a resulting intent classifier will result in some queries from the original `weather` category to be classified to the newly-added `get_weather` category because these two categories are semantically similar. Table 1 illustrates several example colliding intents and associated queries. Our approach addresses these collisions by detecting their prevalence and quantifying their impact automatically, aiding developers in improving the quality of their datasets and scope of their dialogue systems.

Because the notion of semantic overlap can differ from category to category and dataset to dataset, we observe several classes of relationships among colliding intent categories in practice. In particular, intent collisions can be *simple-pairwise*, *transitive*, or *hierarchical*. In the simple-pairwise case, two intents collide with each other *only*, and not with any other intent in either dataset. However, we also observe transitivity within intent classes. Figure 1 illustrates example utterances within intent classes a, b, and c, where all intent classes are transitively related to one another in a cycle.

Lastly, we observe non-transitive hierarchies among colliding intents. In this case, a broad intent category from one dataset can collide with two or more intent categories that do not relate to each other. Figure 2 shows a hypothetical intent class x consisting of general banking queries, including balance inquiries and transfer requests, and classes y and z consist solely of balance inquiry and transfer requests, respectively. Here, because class x is more broad than y and z, each of y and z collide with x, but y and z do not collide with each other. Our approach can help developers reveal

Dataset	# Intents	# Collisions
ACID	175	36
Clinc-150	150	158
MTOP	113	60
Banking-77	77	25
HWU	64	103
New	58	5
MetalWOz	51	80
DSTC-8	34	67
ATIS	26	7
Outlier	10	9
Snips	7	20
Jobs640	1	0
Talk2Car	1	0
Total	767	570

Table 2: Number of intents with collisions. A total of 570 intents have at least one collision.

such cases when managing datasets, and we consider these collision relationships in the creation of our Redwood dataset.

3.3 Approaches

We introduce two approaches for detecting collisions: *Classifier Confusion* and *Data Coverage*.

Classifier Confusion. A column of a confusion matrix charts the distribution of predictions of a classifier for data in a particular category. We call such a distribution the *classification distribution*. We adapt this notion for our first collision detection approach, which identifies a candidate intent A to collide with $B \in \mathcal{C}$ if a classifier model trained on dataset $\mathcal{C}$ produces a *classification distribution d* such that $\frac{max(d)}{sum(d)} > \tau$, where τ is a threshold set by the developer. We call this ratio the *classifier collision score*.

Data Coverage. We define the *coverage* of one intent B over another intent A as

$$\text{Coverage}(A, B) = \frac{1}{|B|} \sum_{b \in B} \max_{a \in A} sim(a, b).$$

Here, $sim(a, b)$ computes the similarity between two phrases a and b (for instance, $sim(a, b)$ could be the cosine similarity between two phrase embeddings or the Jaccard similarity between n-*gram* sets). The coverage metric can be used to detect if two intents collide using a threshold rule. In other words, A and B collide if Coverage$(A,B) > \kappa$, where κ is a threshold chosen by the developer. We call the coverage metric the *coverage score*.

```
[
    {
        "source": "clinc150",
        "intent": "weather"
        "name": "clinc150__weather"
        "collisions": [
            "snips__get_weather",
            "hwu__get_weather",
            "mtop__get_weather",
            "metalwoz__weather_check",
            "dstc8__GetWeather"
        ]
    },
    {
        "source": "hwu",
        "intent": "general_praise",
        "name": "hwu__general_praise",
        "collisions": [
            "acid__st_thank_you",
            "clinc150__thank_you"
        ]
    },
    ...
]
```

Figure 3: Example entries in the graph-like collision meta-dataset, showing collisions for *Clinc-150*'s weather intent and *HWU*'s general_praise intent.

3.4 Datasets

We evaluate the effectiveness of our intent collision approaches using several indicative datasets. These datasets can be roughly grouped into three categories: (1) intent classification datasets like *Clinc-150* (Larson et al., 2019b), *Banking-77* (Casanueva et al., 2020), *ACID* (Acharya and Fung, 2020), *Outlier* (Larson et al., 2019a), and *New* (this work; a corpus that was crowdsourced in a manner similar to Larson et al. (2019b) and Larson et al. (2019a)); (2) joint slot-filling and intent classification or semantic parsing datasets like *ATIS* (Hemphill et al., 1990; Hirschman et al., 1992, 1993; Dahl et al., 1994), *Snips* (Coucke et al., 2018), *HWU* (Liu et al., 2019), and *MTOP* (Li et al., 2021); and (3) turn-based dialog datasets like *DSTC-8* (Kim et al., 2019) and *MetalWOz* (Lee et al., 2019). We only consider the initial queries in the turn-based *DSTC-8* and *MetalWOz*, and discard all subsequent dialog turns.

Queries in these datasets span a wide range of topic domains, including banking and personal finance (*Banking-77* and *Outlier*) and insurance (*ACID*); other datasets cover a wide array of topic domains, such as *Clinc-150* and *HWU*, which cover smart home, automotive, travel, banking, cooking, and others. Since we are concerned with detecting colliding intents, we do not consider any slot annotations, and we use only the first turns from

the multi-turn dialog datasets. In addition, we also use the *Jobs640* (Califf and Mooney, 1997) and *Talk2Car* (Deruyttere et al., 2019) datasets, which, although not originally designed for intent classification tasks, are categorized in a way that admit consideration as single-intent classification for our purposes. Table 2 summarizes these datasets.

The Collision Meta-Dataset We constructed a graph-like dataset that indicates the collision relationships between intents. To build this dataset, we reviewed all intents from all of the datasets listed in Table 2 to check for collisions between other intents. We developed a ground truth set of tuples indicating whether two intents collide among these datasets. Figure 3 shows the structure of the intent collision meta-dataset, and Table 2 displays the number of collisions that occur relative to each individual dataset. The meta-dataset includes the three types of collisions defined in Section 3.2.

3.5 Experimental Evaluation

Implementation Details. We evaluate our intent collision detection methods on our newly-created collision meta-dataset. For evaluating the classifier confusion approach, we train a multi-class intent classifier on each individual dataset (except the single-intent datasets) and then run inference on all other intents from the other datasets. We compute and report the classifier confusion score for each run. In our experiments, we use a linear SVM classifier with bag-of-words feature representations.
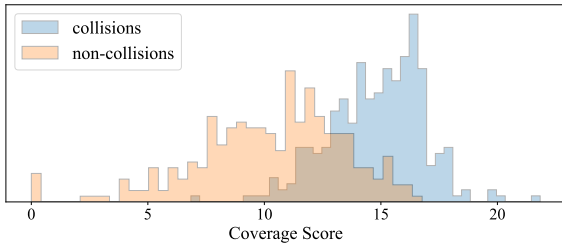
For evaluating the data coverage approach, we first sample[7] a nearly equal number of colliding and non-colliding intent pairs from the collision meta-dataset. We then compute the coverage scores for the selected pairs using several sentence representation and similarity metrics. We use the SBERT library's SBERT-NLI and SBERT-miniLM sentence embedders (Reimers and Gurevych, 2019) along with cosine similarity. Additionally, we also use *n*-gram-based similarity, defined as

$$sim(a,b) = \frac{1}{N}\sum_{n=1}^{N} \frac{|n\text{-grams}_a \cap n\text{-grams}_b|}{|n\text{-grams}_a \cup n\text{-grams}_b|}$$
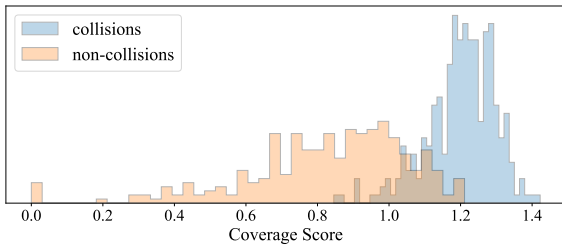
where a and b are queries from two intents, and $N = 3$ in our experiments.

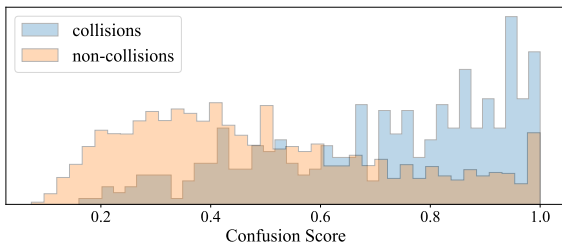For both the data coverage and classifier confusion experiments, we only consider intents that

---

[7] Sampling avoids combinatorial explosion of possible intent pairs.

(a) SBERT-NLI Coverage Score

(b) Mini-LM Coverage Score

(c) SVM classifier confusion.

Figure 4: Data coverage and classifier confusion score distributions for various intent collision detection approaches.

Approach	Coverage AUC	Confusion AUC
SBERT-NLI	0.898	—
token	0.931	—
SBERT-miniLM	0.963	—
SVM-based	—	0.756

Table 3: AUC metrics for each intent collision detection approach.

method's ability to distinguish collisions versus non-collisions; an AUC score of 1.0 means perfect separability between collisions and non-collisions, while an AUC score of 0.5 means a method is unable to distinguish between colliding and non-colliding intents.

3.6 Results

Data Coverage. Figure 4 charts coverage scores and confusion scores for various approaches. In Figure 4 (a) and (b), the coverage approaches tend to return higher coverage scores for non-collisions and lower coverage scores for collisions, which aligns with our expectations given our definition of the coverage metric and assuming the similarity metric used in the coverage computation is effective. The AUC scores allow us to quantitatively judge the performance of the various coverage-based approaches: in Table 3, the SBERT-miniLM embedding method yields the highest AUC score, and interestingly the n-gram-based coverage method performs second best, with the SBERT-NLI embedding method in third.

Classifier Confusion. Figure 4 (c) charts classifier confusion scores for the SVM-based classifier confusion approach. Our results demonstrate that actual intent collisions typically yield high classifier confusion scores, while non-collisions yield lower confusion scores. Visually, however, Figure 4 (c) seems to indicate that that the classifier confusion approach is less effective than the coverage-based approaches. This is made more apparent by the AUC score in Table 3. We note that the data coverage and classifier confusion AUC scores are not directly comparable as they use different evaluation settings. Nonetheless, the difference in performance scores does lead us to conclude that the data coverage approach is more effective.

In sum, these experimental results demonstrate that the two intent collision detection approaches introduced here are effective in detecting collisions

have at least 10 queries. For the collision detection experiments, we used all 285 collision pairs and sampled 300 non-colliding pairs since there are substantially more non-colliding pairs. The classifier confusion approach does not compare intents in a pairwise manner, and instead compares a dataset (i.e., a classifier trained on a dataset) against a single intent at a time. We run a classifier on all multi-intent datasets, which yielded a total of 400 collision pairs and 6,802 non-collision pairs for the classifier confusion experiments.

Metrics. While in actual application settings, a user may wish to use thresholds for τ and κ (defined earlier in Section 3.3) to determine whether intents collide, we evaluate both classifier confusion and coverage methods in a threshold-free manner using the AUC score. (In practice, values for τ and κ could be set by the practitioner via cross-validation or by using the meta-dataset provided in this work to set optimal thresholds for their application.) The AUC score allows us to judge each

Original Dataset	Original Intent	Sample
HWU	alarm_remove	*remove the alarm set for 10pm*
Clinc-150	reminder_update	*set a reminder for me to take my meds*
MTOP	get_weather	*should i wear a raincoat tuesday*
Jobs-640	—	*what systems analyst jobs are there in austin*
Talk2Car	—	*switch to right lane and park on right behind parked black car*
Jobs640	Jobs640	*what systems analyst jobs are there in austin*
Snips	add_to_playlist	*add paulinho da viola to my radio rock song list*
Outlier	hours	*tell me the hours of operation for my bank*
New	balance	*do I have holiday time saved*
DSTC-8	LookupMusic	*I like metal songs can you find me some*
ATIS	ground_service	*i'll need to rent a car in washington dc*
MetalWOz	name_suggester	*I need to find a name for my new cat*
Clinc-150	find_phone	*can you help me find my cell*
ACID	info_amt_due	*what is the current amount due on my account*
Banking-77	terminate_account	*how do I deactivate my account*
Clinc-150	measurement_conversion	*what amount of millimeters are in 50 kilometers*
ACID	info_name_change	*i need to fix my name*
MTOP	play_music	*find me the latest linkin park album*
HWU	audio_volume_up	*just increase the volume a little*
Outlier	balance	*how much oney do i have available*
Vertanen (2017)	—	*why on earth is there cereal in the fridge*
Vertanen (2017)	—	*who are you going to vote for in november*
Vertanen (2017)	—	*do you know where i put my glasses*
Clinc-150	out-of-scope	*what size wipers does this car take*
Clinc-150	out-of-scope	*how long is winter*
Clinc-150	out-of-scope	*are any earning reports due*

Table 4: Sample intents and queries from our *Redwood* dataset, along with the corresponding original dataset and intent (where applicable). Samples are grouped into *in-scope* (top) and *out-of-scope* (bottom).

among real datasets, with the data coverage approach being the stronger of the two.

4 Building the *Redwood* Dataset

With tools addressing the problem of intent collision detection in hand, we now turn our attention to combining the individual datasets from Table 2 together to form a single large-scale intent classification dataset, *Redwood*. This section discusses the construction of *Redwood* and a companion *out-of-scope* evaluation set, and then evaluates several benchmark intent classifiers on the dataset. These datasets and associated evaluations demonstrate the consequences of leaving colliding intents unaddressed, providing a valuable resource for the community to improving intent classification models.

4.1 Data

***In-Scope* Data.** After creating the collision meta-dataset, a natural extension was to combine each dataset together to form *Redwood*. We used the collision meta-dataset to help inform us of which intents could combined, and which intents could stand alone in *Redwood*. In some cases, we removed intents that caused *hierarchical* collisions,

Dataset	N. Samples
Vertanen (2017)	2067
Clinc-150	1200
Total	3267

Table 5: Sources of *out-of-scope* data and number of samples used in *Redwood*'s out-of-scope test set.

as sometimes joining together intents from a hierarchical collision produced an intent that was too broad. We included only those intents that have at least 50 queries, and the resulting *Redwood* consists of 451 total intents and 62,216 queries. Following the terminology used in Larson et al. (2019b), we call these 451 intents *in-scope*.

By way of comparison, we also produced a "naïve" version of *Redwood*, called *Redwood-naïve*, where all the intents from the datasets listed in Table 3.4 were joined together *without* using collision detection or any other method of arbitrating or correcting colliding intents. Like the original *Redwood*, we included only intents that have at least 50 queries, and capped each intent at a maximum of 150 queries so as to avoid drastic class imbalances. *Redwood-naïve* consists of 619 intents and 85,746 total queries.

All versions of *Redwood* were split into train and test splits per intent: 85% training, 15% testing.

***Out-of-Scope* Data.** In contrast to *in-scope*, *out-of-scope* queries are those that do not belong to any of the *in-scope* intents. Considering *out-of-scope* queries in an evaluation of intent classification models is important because such queries occur in production settings, where end users cannot be expected to know the full range of intents when interacting with a conversational AI system. We include a collection of 3,267 *out-of-scope* queries in addition to the *Redwood* corpus. *Redwood*'s *out-of-scope* data originates from the following sources: *Clinc-150* dataset, which itself includes a set of *out-of-scope* queries; and Vertanen (2017), a crowd-sourced dialog dataset from which we use the first dialog turns. We reviewed all candidate *out-of-scope* queries, removing those that were actually *in-scope*. Examples of queries from the *Redwood* dataset are shown in Table 4.

4.2 Benchmark Evaluation

Models. We benchmark intent classification performance using the MobileBERT model (Sun et al., 2020) using the HuggingFace library (Wolf et al., 2020). The MobileBERT implementation uses a softmax function to compute logits to a probability vector p, from which we can obtain confidence scores for each intent. These confidence scores can be used to predict whether a query is in- or out-of-scope, according to a decision threshold t given by

$$\text{decision rule} = \begin{cases} \text{in-scope,} & \text{if } \max(p) \geq t \\ \text{out-of-scope,} & \text{if } \max(p) < t. \end{cases}$$

Such decision rules were used in Hendrycks and Gimpel (2017) and Larson et al. (2019b).

Metrics and Experiments. We measure intent classifier accuracy on in-scope data without considering out-of-scope inputs. We also measure each model's ability to distinguish in-scope and out-of-scope queries by computing the AUC between in- and out-of-scope confidence scores. In this way, we use AUC to measure how separable in- and out-of-scope queries based on their confidence scores without having to select an confidence threshold t. An AUC score of 0.5 (the minimum AUC score) implies the model cannot distinguish in- versus out-of-scope inputs. An AUC of 1.0 indicates the model can perfectly separate inputs.

Training Dataset	In-Scope Accuracy	Clinc OOS AUC	Vertanen OOS AUC
Redwood	0.913	0.921	0.928
Redwood-naïve	0.861	0.909	0.925

Table 6: Model performance of the MobileBERT classifier on *Redwood* and *Redwood-naïve*.

Collisions	0	1	2	3	4	5	6	14
Mean Acc.	0.91	0.80	0.81	0.79	0.81	0.80	0.89	0.57
Size	322	74	42	51	15	11	13	8

Table 7: Accuracy scores on *Redwood-naïve* intents per number of collisions.

4.3 Results

Model performance on *Redwood-naïve* and *Redwood* is shown in Table 6. First, we notice that the intent classifiers perform reasonably well on the in-scope classification task, with MobileBERT classifying queries with 91% accuracy. The models also perform well on the out-of-scope task, and discriminate between in- and out-of-scope queries with AUC scores of 0.921 and 0.928 on the *Clinc-150* and Vertanen (2017) out-of-scope data.

The bottom half of Table 6 presents model performance when trained and tested on *Redwood-naïve*. In this case, model performance is substantially worse than models trained on the carefully-crafted *Redwood* dataset, confirming our hypothesis from Section 3.1 that model performance suffers if trained on data with colliding intents.

We drill deeper into the impact of intent collisions on models trained on *Redwood-naïve* in Table 7 which charts per-intent accuracy based on the number of other intents that collide with that intent. This table groups intents based on the number of collisions, and we see that on average, intents with no collisions exhibit higher accuracy than intents with collisions. In general, colliding intents lead to degraded accuracy: intents with one or more collisions have accuracy of around 10 or more points lower than the no-collision group, with the exception of the 6-collision group. The average accuracy of the 6-collision group on *Redwood-naïve* is indeed surprising, and we posit that the MobileBERT model—a high-capacity transformer model—can learn the nuances of each individual intent, even if they do semantically collide.

5 Conclusion and Future Work

This paper introduces the task of intent collision detection when constructing or updating an intent

classification model's dataset to incorporate additional intents. Using 13 individual datasets, we constructed a meta-dataset to track intent collisions between the datasets, and then introduced and evaluated two intent collision detection techniques and found that both perform effectively at the collision detection task. To help measure and address this problem, we constructed *Redwood*, a large-scale intent classification dataset consisting of 451 intents and over 60,000 queries. We used *Redwood* to benchmark several intent classification models on the task of in-scope query prediction and out-of-scope detection, The new *Redwood* dataset is the largest publicly available intent classification benchmark, in terms of number of intents, and will be made publicly available. Future work will include annotating slots to extend Redwood to joint intent classification and slot-filling, and it is likely that new tools will have to be developed for doing so. Additionally, using the collision detection methods introduced in this paper, *Redwood* can be periodically updated with new intents whenever other new intent classification datasets are published.

Acknowledgements

References

Shailesh Acharya and Glenn Fung. 2020. Using optimal embeddings to learn new intents with few examples: An application in the insurance domain. In *Proceedings of the KDD 2020 Workshop on Conversational Systems Towards Mainstream Adoption (KDD Converse 2020)*.

Mary Elaine Califf and Raymond J. Mooney. 1997. Relational learning of pattern-match rules for information extraction. In *CoNLL97: Computational Natural Language Learning*.

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*.

Christopher Clarke, Joseph Peper, Karthik Krishnamurthy, Walter J. Talamonti, Kevin Leach, Walter S. Lasecki, Yiping Kang, Lingjia Tang, and Jason Mars. 2022. One agent to rule them all: Towards multi-agent conversational ai. *ArXiv*, abs/2203.07665.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, Maël Primet, and Joseph Dureau. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *CoRR*, abs/1805.10190.

Deborah A. Dahl, Madeleine Bates, Michael Brown, William Fisher, Kate Hunicke-Smith, David Pallett, Christine Pao, Alexander Rudnicky, and Elizabeth Shriberg. 1994. Expanding the scope of the ATIS task: The ATIS-3 corpus. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.

Thierry Deruyttere, Simon Vandenhende, Dusan Grujicic, Luc Van Gool, and Marie-Francine Moens. 2019. Talk2Car: Taking control of your self-driving car. 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)*.

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*.

Dan Hendrycks and Kevin Gimpel. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of International Conference on Learning Representations (ICLR)*.

L. Hirschman, M. Bates, D. Dahl, W. Fisher, J. Garofolo, D. Pallett, K. Hunicke-Smith, P. Price, A. Rudnicky, and E. Tzoukermann. 1993. Multi-site data collection and evaluation in spoken language understanding. In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.

Lynette Hirschman, Madeleine Bates, Deborah Dahl, WIlliam Fisher, John Garofolo, Kate Hunicke-Smith, David Pallett, Christine Pao, Patti Price, and Alexander Rudnicky. 1992. Multi-site data collection for a spoken language corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*.

Seokhwan Kim, Michel Galley, Chulaka Gunasekara, Sungjin Lee, Adam Atkinson, Baolin Peng, Hannes Schulz, Jianfeng Gao, Jinchao Li, Mahmoud Adada, Minlie Huang, Luis Lastras, Jonathan K. Kummerfeld, Walter S. Lasecki, Chiori Hori, Anoop Cherian, Tim K. Marks, Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, and Raghav Gupta. 2019. The eighth dialog system technology challenge. In *NeurIPS Workshop: Conversational AI: Today's Practice and Tomorrow's Potential*.

Stefan Larson, Anish Mahendran, Andrew Lee, Jonathan K. Kummerfeld, Parker Hill, Michael A. Laurenzano, Johann Hauswald, Lingjia Tang, and Jason Mars. 2019a. Outlier detection for improved data

quality and diversity in dialog systems. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*.

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. 2019b. 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)*.

Sungjin Lee, Hannes Schulz, Adam Atkinson, Jianfeng Gao, Kaheer Suleman, Layla El Asri, Mahmoud Adada, Minlie Huang, Shikhar Sharma, Wendy Tay, and Xiujun Li. 2019. Multi-domain task-completion dialog challenge. In *Dialog System Technology Challenges 8*.

Haoran Li, Abhinav Arora, Shuohui Chen, Anchit Gupta, Sonal Gupta, and Yashar Mehdad. 2021. MTOP: A comprehensive multilingual task-oriented semantic parsing benchmark. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume (EACL)*.

Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. 2019. Benchmarking natural language understanding services for building conversational agents. In *Proceedings of the Tenth International Workshop on Spoken Dialogue Systems Technology (IWSDS)*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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)*.

Jacob Solawetz and Stefan Larson. 2021. LSOIE: A large-scale dataset for supervised open information extraction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume (EACL)*.

Congzheng Song, Alexander Rush, and Vitaly Shmatikov. 2020. Adversarial semantic collisions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Zhiqing Sun, Hongkun Yu, Xiaodan Song, Renjie Liu, Yiming Yang, and Denny Zhou. 2020. MobileBERT: a compact task-agnostic BERT for resource-limited devices. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Shyam Upadhyay, Manaal Faruqui, Gokhan Tur, Dilek Hakkani-Tur, and Larry Heck. 2018. (almost) zero-shot cross-lingual spoken language understanding. In *Proceedings of the IEEE ICASSP*.

Keith Vertanen. 2017. Towards improving predictive aac using crowdsourced dialogues and partner context. In *ASSETS '17: Proceedings of the ACM SIGACCESS Conference on Computers and Accessibility*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*.

Weijia Xu, Batool Haider, and Saab Mansour. 2020. End-to-end slot alignment and recognition for cross-lingual NLU. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Hanlei Zhang, Xiaoteng Li, Hua Xu, Panpan Zhang, Kang Zhao, and Kai Gao. 2021. TEXTOIR: An integrated and visualized platform for text open intent recognition. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*.

Yiwen Zhang, Caixia Yuan, Xiaojie Wang, Ziwei Bai, and Yongbin Liu. 2022. Learn to adapt for generalized zero-shot text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*.