Exploring Description-Augmented Dataless Intent Classification

Ruoyu Hu[1,*], **Foaad Khosmood**[2] and **Abbas Edalat**[1]
[1]Department of Computing, Imperial College London, UK
[2]California Polytechnic State University, USA
{ruoyu.hu18, a.edalat}@imperial.ac.uk
foaad@calpoly.edu

Abstract

In this work, we introduce several schemes to leverage description-augmented embedding similarity for dataless intent classification using current state-of-the-art (SOTA) text embedding models. We report results of our methods on four commonly used intent classification datasets and compare against previous works of a similar nature. Our work shows promising results for dataless classification scaling to a large number of unseen intents. We show competitive results and significant improvements (+6.12% Avg.) over strong zero-shot baselines, all without training on labelled or task-specific data. Furthermore, we provide qualitative error analysis of the shortfalls of this methodology to help guide future research in this area.

1 Introduction

Task-oriented dialogue systems (TODS) by design, aid the user in accomplishing tasks within specific domains, and can have a wide range of applications from shopping (Yan et al., 2017) to healthcare (Wei et al., 2018; Valizadeh and Parde, 2022). Modular TODS (Wen et al., 2017) will typically contain an intent classification component (Louvan and Magnini, 2020; Chen et al., 2019; Su et al., 2022) used by a dialogue manager to determine the appropriate task the user intends to complete. In recent years, neural-based models using supervised training have reached state-of-the-art on many natural language processing tasks, including intent classification. However, supervised learning methods require human-labelled data for a predefined set of intents, which may be time-consuming and labour-intensive to acquire (Xia et al., 2018), and may have poor scalability if new intents are added, or task definition changed. An early approach to tackle this problem is *dataless intent classification* (Chang et al., 2008; Song and Roth, 2014) which aimed to leverage the pairwise similarities between semantic representations of utterances and intent classes to perform classification without reliance on human-labelled data. However, this approach relies heavily on the quality of semantic representations (Chang et al., 2008). In recent years, successful *zero-shot intent classification* approaches (Liu et al., 2019; Yan et al., 2020; Yin et al., 2019) have received greater attention, whereby learning conducted using labelled examples of a subset of *seen* intent labels is transferred to *unseen* intents. However, these methods still require human-labelled data, and tend to bias towards seen intents, with the number of unseen intents also generally much lower than seen intents (Liu et al., 2022; Zhang et al., 2022).

In this work, with the significant recent advancements in the quality of text embedding models (Muennighoff et al., 2023), we explore the potential for dataless intent classification methods using a number of recent state-of-the-art text embedding models. We introduce several approaches for generating intermediate textual representations for intents, most notably using intent label descriptions, and formalise our methodology. We perform extensive evaluation of our methods, including scenarios with large numbers of intents from different domains, using three commonly used intent classification datasets. We summarise our contributions as follows:

- We introduce a new scheme for generating intent descriptions with an aim to minimise reliance on human expert input.
- We show that our intent descriptions yield significant improvements over label tokenization through extensive evaluation.
- We introduce an approach utilising utterance paraphrasing and masking which yields further improvements and show this is consistent across a range of models.
- We aggregate and explore the potential of a multitude of current SOTA text embedding

*Corresponding author

Proceedings of the 6th Workshop on NLP for Conversational AI (NLP4ConvAI 2024), pages 13–36
August 16, 2024 ©2024 Association for Computational Linguistics

models for dataless classification.

- We extensively evaluate our methodology on four commonly used intent classification datasets and report on the results.
- We provide qualitative error analysis aimed at guiding future work.

2 Related Works

2.1 Generalized Zero-Shot Learning

Zero-shot learning (ZSL) (Yin et al., 2019) aims to leverage learning previously performed on labelled examples from seen tasks to unseen tasks, of which there are no labelled examples available for supervised training. ZSL has seen increasing popularity in the domain of intent classification (Liu et al., 2019; Yan et al., 2020) in recent years, whereby models are trained on a subset of intent labels and evaluated on another disjoint subset of intent labels. In more recent years, the concept of generalized zero-shot learning (GZSL) has seen an increase in prominence in the domain, in which the performance on both seen and unseen classes are considered in tandem (Zhang et al., 2022; Lamanov et al., 2022). Several GZSL approaches learn a label prototype space during training, which is transferred to unseen classes through methods such as interclass relationship modelling (Zhang et al., 2021) and prototype adaptation (Zhang et al., 2022). Approaches such as (Lamanov et al., 2022) encode the utterance and labels in a sentence-pair setup, with template-based lexicalisation of labels used as class prototypes. Other approaches exist that use label prototypes as centroids in Gaussian mixture models trained on seen class utterances (Yan et al., 2020; Liu et al., 2022). An issue that can occur with GZSL is biased towards seen classes (Zhang et al., 2022), which can lead to significantly lower performance on unseen classes. It is also difficult to see the efficacy of transfer to a large number of diverse unseen classes, as the number of unseen classes in evaluation is also typically much smaller than the number of seen classes.

2.2 Dataless Classification

Dataless text classification (Chang et al., 2008) is defined as tackling text classification without prior training on any labelled data. Generally regarded as a precursor to zero-shot text classification, this approach typically leverages sentence representations without any training on labelled data, by comparing the semantic representations between a sentence and that of the intent classes (Song and Roth, 2014). (Zha and Li, 2019) utilises “seed" words associated with each intent class to further contextualise the intent class representation, as a single word may often be insufficient to encapsulate the meaning of the class (Chen et al., 2015). Some approaches further leverage class hierarchy to augment classification performance (Li et al., 2016; Popov et al., 2019).

3 Methodology

3.1 Problem Definition

Let $\mathcal{C}$ be a set of intents supported by a task-oriented dialogue system, $\mathcal{U} = \bigcup\{\mathcal{U}_c\}_{c\in\mathcal{C}}$ defines the set of all user utterances, $\mathcal{U}_c = \{u_i\}_{1\leq i\leq n_c}$ is the set of utterances belonging to intent class c. The model undergoes no task-specific training and is tasked with making an intent prediction $\hat{y}_i$ for a previously unseen utterance u_i at inference time. We follow the paradigm set by previous works in dataless text classification (Chang et al., 2008; Song and Roth, 2014) to conduct nearest-neighbour classification over the sentence embedding space. For a given utterance u_i, an encoder $\mathbf{h}(\cdot)$ and a set of class label representations $\{l_c\}_{c\in\mathcal{C}}$, we make a prediction $\hat{y}_i$ as follows:

$$\hat{y}_i = \arg\max_c s(\mathbf{h}(u_i), \mathbf{h}(l_c)) \tag{1}$$

where $s(\mathbf{u}, \mathbf{v}) = \mathbf{u}\cdot\mathbf{v}/||\mathbf{u}||_2||\mathbf{v}||_2$ is the cosine similarity between two vectors.

In order to conduct nearest-neighbour classification using intent labels, we require an intermediate representation, or prototype, which encapsulates to some degree the meaning of a class (Zha and Li, 2019), from which we can obtain a suitable embedding. A commonly used approach in dataless classification is to use the labels (Chang et al., 2008).

3.2 Label Tokenization

A class prototype is obtained by tokenizing intent labels directly, inserting spaces and replacing character separators, i.e.

```
AddToPlaylist → Add To Playlist
oil_change_how → Oil Change How
```

However, this approach depends on the descriptiveness of the original intent labels, which can vary significantly between datasets and tasks. As such, we propose an additional step to produce intent label *descriptions* which we hypothesise can (1) better align the semantic representation with the characteristics of the class and (2) provide more consistent performance across datasets or approaches without requiring in-task data, which previous works (Lamanov et al., 2022) have shown could improve performance over purely using tokenized labels.

3.3 Our Approach

3.3.1 Intent Description

Our objective is to produce a brief description of the intent expressed by the user in a given utterance, while ensuring the process requires minimal expert human effort so as to remain scalable for large numbers of intent classes. Rather than producing a general description of the intent (Gao et al., 2023), we formalise our template for producing intent descriptions with the two following constraints:

Label Preservation The resulting intent description must contain tokens from the original intent label i.e. `car_rental` → `User wants to rent a car`, or replace with an appropriate word (lexical cognates, synonyms etc.).

Format Consistency Descriptions should be written in the declarative form, beginning with either `"User is [asking|saying]"`, or `"User wants [to]"`, and aim to introduce minimal extraneous tokens in a similar manner to abstractive summarization (De Raedt et al., 2023). Our approach differs from the template-based approach in (Lamanov et al., 2022) in that we use exclusively the declarative form in writing our descriptions to maintain consistency across intent classes and datasets. Example descriptions can be seen in Table 1, more examples can be found in Appendix I. We examine the robustness of our approach in Section 6.

In our experimentation (Section 4), our intent descriptions added on average 6.6 tokens to the tokenized intent labels (1.9 → 8.5), with 98.3% of descriptions containing at least one of the label tokens in exact form, and 82.7% of all label tokens preserved.

3.3.2 Utterance Paraphrasing

The diversity of user utterances for any given intent can pose a challenge as intents may not be obvious (Mueller et al., 2022). We hypothesise that a format consistency constraint over the user utterance can benefit dataless intent classification performance. Previous works primarily focused on utterance paraphrasing as a means of data augmentation (Kumar et al., 2019; Jolly et al., 2020; Sahu et al., 2022) or to reduce overfitting (Dopierre et al., 2021). Our approach leverages inference-time paraphrasing to enforce a weaker degree of our intent descriptions' format consistency constraint on user utterances. Given a paraphraser model $\mathbf{p}(\cdot)$ we compute a sentence embedding of the paraphrased utterance $\mathbf{p}(u_i)$:

$$P_{u_i} = \mathbf{h}(\mathbf{p}(u_i)) \tag{2}$$

We leverage a 1.6B StableLM model[1] (Bellagente et al., 2024) to generate a single paraphrase for each utterance. Our selection was based on said model being the top-performing model under 2B parameters on the Open LLM Leaderboard (Beeching et al., 2023) as of the time of writing. We additionally experimented with 1.6B Zephyr (Tunstall et al., 2023) and 1.3B Phi-1.5 (Li et al., 2023a) models but found no significant difference on our task. Example templates and further details are shown in Appendix A. The mean cosine similarity between the paraphrases and the original utterances across 4 intent classification tasks and 12 embedding models is 0.89 ± 0.06.

Label	Description
`abbreviation`	"user is asking what an abbreviation stands for or means"
`flight_no`	"user is asking about a flight number"
`AddToPlaylist`	"user wants to add a song to a playlist"
`food_last`	"user wants to know how long a food lasts
`maybe`	"user is expressing uncertainty"

Table 1: Example descriptions for intent labels from each of the datasets (Section 4.1) used in our experimentation.

3.3.3 Label Entity Overlap & Masking

We note that sentence embeddings tended to capture the topic and entity information rather than the associated action, which can lead to misclassifications in the event that two or more intent classes share entities (i.e. `AddToPlaylist` and `PlayMusic` can both refer to songs as their objects). To tackle this, we introduce a masking

[1] `https://huggingface.co/stabilityai/stablelm-2-1_6b-chat`

Algorithm 1 Utterance Masking Procedure

```
Given user utterance u_i = {u_{i,1}, ... u_{i,t}}
T_i ← DependencyParser(u_i)
procedure MASKTREE(T)
    n ← root(T)
    if relation(n) is obj then
        n ← [MASK]
        DROP children(n)
    else
        for u_{i,j} in children(n) do
            MASKTREE(u_{i,j})
        end for
    end if
end procedure
```

component which given user utterance u_i masks spans containing the object of said utterance, identified through dependency parsing[2] (de Marneffe and Manning, 2008; Schuster and Manning, 2016), to produce m_i. m_i is then weighted to form the masking component:

$$M_{u_i} = \mathbf{h}(m_i) \times Overlaps(u_i, k) \times \mathbb{1}_{masked} \quad (3)$$

where $Overlaps(u, k)$ denotes whether there is likely entity overlap in the top k candidate intents by similarity and $\mathbb{1}_{masked}$ is whether there exists a masked version of the original sentence. We did not find significant differences in performance for $k > 3$, and thus we use $k = 3$ for all our experiments.

Masking Algorithm 1 illustrates the masking procedure which identifies and masks object spans in the utterance. We define such object spans as subtrees within the dependency tree in which a parent node has any of `{dobj, pobj, ccomp}` relations. We note that object relations are not always present in the dependency tree, in such cases masked representations are not used. From our experiments, some degree of masking was performed for 97.29% of utterances from the ATIS dataset, 98.04% of SNIPS, 90.88% of CLINC and 84.24% of MASSIVE. We show an example of this procedure in Appendix B.

Entity Overlap For each intent, we predict a set of entities $\mathbf{e}_c = \{e_{c_1}, \ldots, e_{c_k}\}$ from the intent description that may describe the object of said class. As such, entities are defined at problem definition and can be modified alongside intent descriptions when they are added/removed. We precompute an overlap matrix $Overlap$ where

$$Overlap[i, j] = \begin{cases} 1 & \mathbf{e}_i \cap \mathbf{e}_j \neq \varnothing \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

At inference time, we compute overlaps for classes with top k embedding similarities for an utterance u_i. Given a similarity vector $s_i = \{s_{i,1}, \ldots, s_{i,c}\}_{c=|\mathcal{C}|}$ of embedding similarities between utterance embedding $\mathbf{h}(u_i)$ and intent description embeddings $\mathbf{h}(l_c)_{c \in \mathcal{C}}$, we compute $Top_k(u_i)$ as the top k classes with similarity scores sorted in descending order. We then compute pairwise overlap for all pairs in $Top_k(u_i)$ as follows:

$$Overlaps(u_i, k) = \bigcup_{m,n \in Top_k(u_i), m \neq n} Overlap[m, n] \quad (5)$$

We note that future work could explore expansion of the definition of relevant entities to each intent class, as the current solution relies on the quality of intent descriptions and only covers the most likely entities across an entire class, a more dynamic inference-time solution that determines overlap based on candidate classes would be desirable.

3.4 Combined Sentence Representation

We formulate the final representation of the user utterance within the embedding space as the sum of the original utterance embedding with the paraphrasing and masking components:

$$h_i = \mathbf{h}(u_i) + P_{u_i} + M_{u_i} \quad (6)$$

$$\hat{y}_i = \arg\max_c s(h_i, \mathbf{h}(l_c)) \quad (7)$$

4 Experiments

4.1 Datasets

We evaluate our methods on four commonly used English task-oriented dialogue (TOD) system intent classification datasets, covering a diverse range of number of intents (7-150) and domains (up to 18). (1) **ATIS** (Hemphill et al., 1990) is an English air-travel information system dataset containing 18 intent classes. For comparison, we follow previous works (Zhang et al., 2022) in filtering out intent classes containing fewer than 5 examples.

[2] We leverage an off-the-shelf dependency parser, `en_core_web_trf` from `Spacy` **url**: `https://spacy.io/models/en`

(2) **SNIPS-NLU** (Coucke et al., 2018) contains 7 intent classes, totalling 14,484 utterances. (3) **CLINC** (Larson et al., 2019) is a dataset for out-of-scope intent classification, with 150 intents and 22,500 utterances spanning 10 domains. (4) **MASSIVE** (FitzGerald et al., 2023) is a multilingual spoken language understanding dataset containing 60 intents across 18 domains, we select the 16,521 instances from the `en-US` split of the dataset for our experiments. As our method does not involve fine-tuning on task-specific data, we consider *entire* datasets to consist of unseen data for evaluation[3].

4.2 Models

We select 11 models from the Massive Text Embedding Benchmark (MTEB) (Muennighoff et al., 2023) that are in the top 20 at the time of writing[4]. Our selections are based on the following criteria: (1) the model weights must be released (2) documentation of training methods and experimentation details must be readily available. Additionally, owing to computational limits[5], we only consider models up to 3GB in size. Our final selection of 11 models can be largely grouped into 4 families of models: **InstructOR** (Su et al., 2023), **E5** (Wang et al., 2022), **GTE** (Li et al., 2023b) and **BGE** (Xiao et al., 2023). More details on selected models are provided in Appendix C.

We report results in Section 5 for all E5, GTE and BGE models using averaged token embeddings as sentence representations. We additionally compare model performances against a commonly used embedding model in OpenAI's `text-embedding-ada-002` (Neelakantan et al., 2022) which we refer to in our tables as 'Ada-002'. We also investigated the generation of synthetic examples as intent prototypes (Appendix I) but did not find significant improvements over our approach using intent descriptions (Appendix J).

[3]We make our code and datasets publicly available and can be found at https://github.com/ruoyunlp/dataless-intent-classification

[4]November-December 2023. We note our top-performing selected models are still competitive with current top-performing models from MTEB fitting our criteria as of May 2024

[5]All experiments conducted using a single 9GB GPU

5 Results

5.1 Baselines and Terminology

We compare the performance of our methods against several unknown intent classification methods previously detailed in Section 2. Here we clarify the terminology used henceforth to refer to these methods in our results. We refer to scores on unseen intent labels reported by (Zhang et al., 2021) as **ICR**, (Yan et al., 2020) as **SEG**, (Liu et al., 2022) as **ML-SEG**, dataless approach trained using original data from (Lamanov et al., 2022) as **TIR**$_{Orig}$ and likewise **TIR**$_{Syn}$ for training on synthetic data. We refer to the results of the adapted method of (Gidaris and Komodakis, 2018) reported in (Zhang et al., 2022) as **CosT** and the reported main results as **LTA**. We refer to the best-performing model of a similar size to our selection from (Gretz et al., 2023) as **TTC**$_D$.

5.2 Metrics

Following from previous works (Zhang et al., 2022; Lamanov et al., 2022), we report Accuracy and Macro-F1 scores for intent classification on each of the datasets, in addition, we also compute the average of Accuracy and F1 score for direct comparison between our methods similar to (Gritta et al., 2022). We show macro-F1 only for MASSIVE in Table 2 for comparison's sake as the previous work (Gretz et al., 2023) did not report Accuracy scores. Full results for each of our approaches including Accuracy scores are shown in Table 9.

5.3 Methods using Tokenized Labels

Despite a lack of task-specific fine-tuning, models using tokenized intent labels generally performed comparably to most of the baselines on unseen intents. The best-performing model (BGE$_{Large}$) outperforms baseline scores for ICR (+9.13 Mean), SEG (+10.21 Mean) and ML-SEG (+3.14 Mean), TIR$_{Syn}$ (+13.60 Mean), TIR$_{Orig}$ (+4.55 Mean) and TTC$_D$ (+0.31 F1). BGE$_{Large}$ outperforms CosT on all datasets; however, it also significantly underperforms LTA on all 3 datasets (-16.38 ATIS, -7.49 SNIPS-NLU, -1.21 CLINC). We note that this approach appears quite sensitive to the model as indicated by the comparatively high standard deviation ($\sigma_{Ovr} = 5.65$) across models.

5.4 Methods using Intent Descriptions

Our method using intent label descriptions yields a significant improvement over using tokenized la-

	Model	AT.	SN.	CL.	MA.	Ovr.
		Mean Acc. & F1			F1	
Baselines	ICR	35.04	-	-	-	-
	SEG	-	69.46	-	-	-
	ML-SEG	-	76.53	-	-	-
	TIR_{Orig}	-	-	68.50	-	-
	TIR_{Syn}	-	-	59.65	-	-
	CosT	45.62	55.28	66.50	-	-
	LTA	60.55	87.16	74.46	-	-
	TTC_D	-	-	-	54.22	-
	Baselines	60.55	87.16	74.46	54.22	69.10
Tokenized Intent Labels	Instr._{Large}	18.72	82.39	62.76	47.62	52.87
	E5-v2_{Base}	20.39	77.13	63.87	45.97	51.84
	E5-v2_{Large}	26.64	69.99	60.40	46.83	50.97
	mE5_{Large}	22.47	59.35	57.34	44.34	45.88
	E5_{Large}	40.57	74.44	69.11	49.78	58.48
	Ada-002	25.98	82.75	66.97	47.90	55.90
	GTE_{Small}	20.75	73.99	68.47	51.90	53.77
	GTE_{Base}	55.66	81.75	70.65	51.44	64.88
	GTE_{Large}	39.78	79.36	69.54	49.08	59.44
	BGE_{Small}	19.50	78.00	70.78	52.43	55.18
	BGE_{Base}	45.74	76.81	73.05	55.89	62.87
	BGE_{Large}	44.17	79.67	73.25	54.53	62.91
Intent Label Descriptions	Instr._{Large}	42.18	85.60	77.25	55.52	65.14
	E5-v2_{Base}	52.44	87.49	70.92	53.73	66.14
	E5-v2_{Large}	52.16	87.31	71.49	55.65	66.65
	mE5_{Large}	60.51	83.88	72.24	56.67	68.32
	E5_{Large}	52.56	88.92	74.88	56.32	68.17
	Ada-002	51.34	89.50	77.81	58.03	69.17
	GTE_{Small}	54.71	84.42	70.20	51.86	65.30
	GTE_{Base}	52.60	86.41	75.10	54.62	67.18
	GTE_{Large}	55.85	86.33	75.83	57.85	68.97
	BGE_{Small}	47.84	85.51	72.03	54.27	64.91
	BGE_{Base}	48.76	88.32	77.61	58.92	68.40
	BGE_{Large}	54.88	89.30	79.08	62.88	71.53
+ Paraphrase and Masking	Instr._{Large}	49.07	89.86	80.17	59.79	69.72
	E5-v2_{Base}	60.93	90.03	75.06	57.81	70.95
	E5-v2_{Large}	48.06	85.56	74.69	58.27	66.64
	mE5_{Large}	57.72	83.36	75.00	57.67	68.43
	E5_{Large}	53.78	91.92	76.27	59.17	70.28
	Ada-002	57.02	90.51	79.73	59.92	71.80
	GTE_{Small}	53.48	88.11	71.50	57.53	67.66
	GTE_{Base}	**64.20**	85.88	75.75	58.41	71.06
	GTE_{Large}	60.63	91.70	78.89	61.63	73.21
	BGE_{Small}	54.16	90.76	75.04	59.11	69.77
	BGE_{Base}	58.69	91.81	79.80	61.98	73.07
	BGE_{Large}	61.04	**92.57**	**81.52**	**65.76**	**75.22**

Table 2: Model performance on 4 intent classification tasks. We show Mean of Accuracy and Macro-F1 scores for ATIS, SNIPS-NLU & CLINC. Macro-F1 is shown for MASSIVE as TTC_D did not report Accuracy. Full results for each dataset are shown in Table 9.

Model	Tok.	Desc.	Comb.
$\text{InstructOR}_{Large}$	64.96	73.19	76.89
E5-v2_{Base}	62.98	71.02	74.58
E5-v2_{Large}	59.75	71.76	73.13
mE5_{Large}	54.23	71.50	72.57
E5_{Large}	64.70	73.65	76.09
Ada-002	66.48	75.35	77.12
GTE_{Small}	65.43	69.38	72.80
GTE_{Base}	68.57	72.35	73.63
GTE_{Large}	66.63	73.57	77.57
BGE_{Small}	68.20	71.11	75.37
BGE_{Base}	69.36	75.28	78.05
BGE_{Large}	69.76	77.15	**79.91**

Table 3: Average model Mean of Accuracy and F1 over SNIPS-NLU, CLINC and MASSIVE datasets using tokenized intent labels (**Tok.**), intent descriptions (**Desc.**) and combined utterance embedding (**Comb.**).

bels (Tables 2 and 3), with an average increase per model of +11.24 overall. This supports our hypothesis (1) (Section 3.2) in that the additional contextualisation added through describing the label via a declarative sentence better encapsulates the semantic information represented by a label. We also note from Table 3 that the standard deviation in performance across models is significantly lower when using descriptions ($\sigma_{Ovr} = 1.98$), supporting our hypothesis (2) that descriptions can improve consistency across models and approaches. Our overall best-performing model (BGE_{Large}) also considerably outperforms the strongest baseline on SNIPS-NLU (+2.14 Mean), CLINC (+4.62 Mean) and MASSIVE (+8.66 F1). We do note that all of our approaches in this setup underperform on the ATIS dataset compared to the baseline, with our overall best-performing approach yielding 60.51 vs 60.55; we provide further insight into possible reasons in Section 7 to help guide future research.

5.5 Methods with Additional Paraphrasing and Masking

Our addition of paraphrase and masked utterance embeddings yields further overall score improvements on average of +3.16 over label descriptions and is consistent across different models (Table 3). Our best-performing model (BGE_{Large}) significantly outperforms previous approaches on all 4 datasets (+0.49 ATIS, +5.42 SNIPS-NLU, +7.06 CLINC, +11.54 MASSIVE). Additionally, our approach outperforms previous work on 9 out of 12 selected models.

E	P	M	O	AT.	SN.	CL.	MA.	Ovr.
x				54.89	89.29	79.08	63.09	71.59
	x			56.03	85.77	78.77	63.35	70.98
		x		30.72	76.76	37.90	33.62	44.75
x	x			56.11	88.83	**81.56**	65.60	73.02
x		x		60.84	92.52	75.56	60.80	72.43
	x	x		60.57	92.19	75.99	62.91	72.92
x	x	x		**61.04**	**92.67**	81.22	<u>65.64</u>	<u>75.14</u>
x		x	x	60.84	92.56	77.36	61.82	73.14
	x	x	x	60.57	92.02	76.86	63.04	73.12
x	x	x	x	**61.04**	<u>92.57</u>	<u>81.52</u>	**65.65**	**75.20**

Table 4: Mean of Accuracy and Macro-F1 on 4 intent classification datasets using a `bge-large-en-v1.5` model. **Setup** denotes whether a component is used in the combined sentence embedding: **E** - utterance embedding, **P** - paraphrasing, **M** - masking, **O** - entity overlap in masking.

6 Ablations

Addition of paraphrasing and masking Table 3 illustrates the mean performance across SNIPS, CLINC and MASSIVE datasets for each model different class prototypes. We note the consistent improvement in performance between tokenized intent labels and our approach using declarative intent descriptions (+7.86 Mean), and the further improvements with added paraphrasing and masking (+10.56 Mean). We omit ATIS from this table as it is significantly unbalanced, the impact of which we explore in Section 7, and its results are already included in Table 2.

Combination of techniques Table 4 demonstrates the performance (mean of accuracy and macro-f1) between different combinations of our techniques using a `bge-large-en-v1.5` model. We observe that the addition of paraphrasing increases performance by an average of +2.06% compared to methods without, supporting our hypothesis (3) that inference-time paraphrasing can benefit dataless intent classification. We observe that masking increases performance by an average of +1.80% and the addition of masked embedding only when entity overlaps are predicted increases performance by +0.32% on average. We perform further ablations over combinations of techniques using other models in Appendix E and note similar behaviour across different models.

Choice of Descriptions To investigate whether our proposed method is sensitive to the choice of intent descriptions, we generate paraphrases of our manually produced descriptions with increasing temperature values and sampled 200 combinations of descriptions for each dataset. Table 5 contains the mean and standard deviations of the macro-f1 scores for each dataset, we report macro-f1 for this ablations experiment due to the severely unbalanced nature of the ATIS dataset towards a single class `flight` (accounting for $\sim 74\%$ of the dataset). Further details on description paraphrase generation and sampling along with examples are provided in Appendix F. Methods using only tokenized intent labels are outperformed by our methods using label descriptions ($+4.51\%$), with further improvements from the addition of paraphrasing and masking ($+8.00\%$). The overall scores per dataset are slightly affected by the choice of intent descriptions, with standard deviations between 1-2% with the exception of the ATIS dataset. Future work could focus on the combination of multiple intent descriptions (via paraphrasing) or description refinement with unsupervised training (Chu et al., 2021; Müller et al., 2022) to further improve robustness to the choice of descriptions.

7 Analysis and Future Work

In-Domain Saturation We visualise the embeddings generated by our best-performing model (BGE_{Large}) on the 4 evaluation datasets using t-SNE (van der Maaten and Hinton, 2008), along with the embedding for the intent label description to gain insight into the source of errors in our approach. Figure 1 shows the distribution of embeddings on the ATIS and SNIPS datasets. In the interest of space, visualisations of CLINC and MASSIVE are shown in Appendix G . We observe a poor alignment on the ATIS dataset between the intent label descriptions (Figure 1a) and utterance embeddings corresponding to each class, possibly explaining the poor performance in general on this dataset across models. We note the single-domain nature of the ATIS dataset, with all utterances relating to air-travel/flight; additionally, we note the significantly imbalanced nature of the ATIS dataset (Nan et al., 2021), with $\sim 74\%$ of utterances belonging to the `flight` class, which is a label that overlaps the domain of the dataset. We hypothesise this may lead to the intent label descriptions being much worse at capturing semantic information distinct to each class. This is supported by analysis of the pairwise embedding similarities of utterances

Setup	ATIS	SNIPS	CLINC	MASSIVE	Overall
Tokenized Intent Labels	40.11	78.74	72.45	54.53	61.46
Intent Label Descriptions	42.00 $\pm$ 3.91	86.97 $\pm$ 2.05	73.77 $\pm$ 1.10	61.12 $\pm$ 1.04	65.97 $\pm$ 2.02
+ Paraphrase & Masking	**46.83** $\pm$ 4.18	**91.21** $\pm$ 1.61	**76.17** $\pm$ 1.14	**63.61** $\pm$ 1.19	**69.46** $\pm$ 2.03

Table 5: Comparison of macro-f1 score across 200 sampled combinations of descriptions for our setups with/without paraphrasing and masking. Note our combined approach outperforms tokenized labels across all datasets.

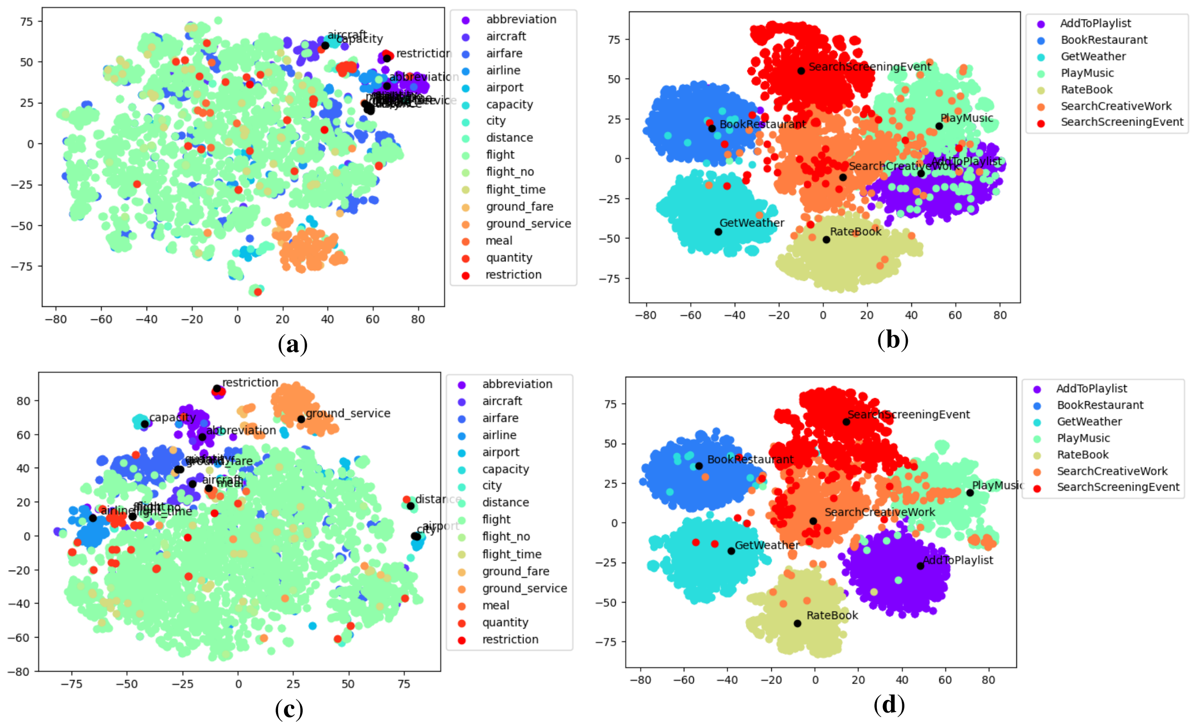

Figure 1: t-SNE (van der Maaten and Hinton, 2008) visualisation of embeddings computed using BGE$_{Large}$, class label description embeddings are shown in black and labelled. (**Row 1**) Embeddings of ATIS (**a**) and SNIPS (**b**), (**Row 2**) Embeddings with Paraphrasing and Masking for ATIS (**c**) and SNIPS (**d**).

belonging to the same class vs utterances belonging to different classes (Table 13) where models' embeddings on the ATIS dataset consistently had lower percentage-difference in embedding similarity between *in*-class and *out*-class, implying more difficulty in distinguishing the utterances using solely embeddings. This issue is mitigated to some degree with our addition of paraphrasing and masking, as the number of misclassifications where there are entity overlaps between classes is reduced on average by 19.19%. We see this visually in Figure 1d as the cluster for each class is more distinct compared to 1b. Errors from classes with overlapping entities in SNIPS are reduced by 29.31%.

Error Analysis We perform qualitative analysis of the remaining errors and identify two categories of commonly occurring errors. (1) *Description Scope:* Our approach utilises a single description for each intent and can work well when an intent concerns a limited number of topics; however, intents such as `meta` and `small_talk` from the CLINC dataset, and `qa` from the MASSIVE dataset can encompass a significantly broader range of topics than other intents within the same dataset. The impact of topical granularity per intent class and the potential for a hierarchical approach to intent classes in a dataless setting can be the focus of future work in this area. (2) *Action Overlap:* Our approach mitigates some errors arising from shared entities across intents through masking. Whilst this has shown success in reducing errors of this nature (i.e. between `PlayMusic` and `AddToPlaylist` from the SNIPS dataset), it is less successful in events where an action is shared across classes, such as `play` from the MASSIVE dataset, and `SearchCreativeWork` and `SearchScreeningEvent` from the SNIPS-NLU dataset. Future work could investigate the potential to decouple the desired *action* and *object*

Dataset	**Top-1**	**Top-3**	**Top-5**	**Top-10**
ATIS	67.70	93.38	96.03	98.10
SNIPS-NLU	89.78	97.13	99.43	100.00
CLINC	77.24	91.71	94.86	97.41
MASSIVE	61.45	81.85	87.79	92.79
Average	74.04	91.01	94.53	97.08

Table 6: Percentage of correct labels within Top-k ranked by embedding similarity per evaluation dataset, averaged across 11 selected models.

(topical information) in utterance embeddings.

Label Candidate Analysis We observed from our results (Table 2) that our approach, despite outperforming strong baselines on ATIS and MASSIVE datasets, still consistently underperforms compared to the same setup on SNIPS-NLU and CLINC. We therefore investigate the position of the correct label when ranking embedding similarities. Table 6 shows the percentage of examples where the correct label is ranked within the top-k by embedding similarity for $k = 1, 3, 5, 10$. We note for erroneous predictions, the correct label is within the Top-3 in 67.11% of cases, 81.89% in Top-5 and 90.94% in Top-10. This implies that our approach can be used to identify candidate intents from a larger set of intents, with a high success rate even for small values of k (i.e. 91.01% Top-3).

Analysis Summary Our proposed approach performs well overall against the strong baseline methods in unseen intent classification; however, it struggles in certain instances with overlaps in intents within the same domain. We identified potential areas for future work to pursue in tackling said issues. The results of our experiments have shown intent label descriptions can perform well as intent prototypes in this problem setting, and that the addition of paraphrasing and masking can further improve performance.

Limitations This approach contains a number of limitations: We have identified issues with the descriptiveness of individual labels earlier in this section, and textual labels may not be readily available for certain datasets, though summarisation methods may be effectively applied to a few user utterances to produce such labels. Our evaluation compares against previous works using scores as reported in their respective papers, further work can be done to replicate their experiments to mitigate any potential risk arising from differences in experimental settings. Future work may also investigate the application of descriptions to tasks outside of intent classification, such as emotion recognition (Rashkin et al., 2019).

8 Conclusion

Dataless classification allows for scaling to a large number of unseen classes without requiring training on labelled, task-specific data. The benefits of such an approach can enhance development of task-oriented dialogue systems in application to data-poor or compute-limited scenarios where supported intents may also change as the system is developed. In this paper, we have explored the potential of current SOTA text embedding models in dataless intent classification settings using three different approaches for representing intent classes and compared our results against strong zero-shot learning baselines. We proposed a method for standardising the generation of intent label descriptions with an aim to minimise the amount of human annotation required to further support scaling to high numbers of intent classes. Our results have shown that description-augmented dataless classification methods can achieve comparable, and sometimes superior performance to zero-shot methods on the task of intent classification.

9 Acknowledgements

We would like to thank Anandha Gopalan for his helpful comments on the paper. Student Ruoyu Hu was funded by UKRI CDT in AI4Health - grant number EP/S023283/1.

References

Edward Beeching, Clémentine Fourrier, Nathan Habib, Sheon Han, Nathan Lambert, Nazneen Rajani, Omar Sanseviero, Lewis Tunstall, and Thomas Wolf. 2023. Open llm leaderboard. `https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard`.

Marco Bellagente, Jonathan Tow, Dakota Mahan, Duy Phung, Maksym Zhuravinskyi, Reshinth Adithyan, James Baicoianu, Ben Brooks, Nathan Cooper, Ashish Datta, Meng Lee, Emad Mostaque, Michael Pieler, Nikhil Pinnaparju, Paulo Rocha, Harry Saini, Hannah Teufel, Niccolo Zanichelli, and Carlos Riquelme. 2024. Stable lm 2 1.6b technical report. *arXiv preprint arXiv:2402.17834*.

Ming-Wei Chang, Lev Ratinov, Dan Roth, and Vivek Srikumar. 2008. Importance of semantic representation: Dataless classification. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 2*, AAAI'08, page 830–835. AAAI Press.

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

Xingyuan Chen, Yunqing Xia, Peng Jin, and John Carroll. 2015. Dataless text classification with descriptive lda. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 2224–2231. AAAI Press.

Zewei Chu, Karl Stratos, and Kevin Gimpel. 2021. Unsupervised label refinement improves dataless text classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4165–4178, Online. Association for Computational Linguistics.

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. *arXiv preprint arXiv:1805.10190*.

Marie-Catherine de Marneffe and Christopher D. Manning. 2008. The Stanford typed dependencies representation. In *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*, pages 1–8, Manchester, UK. Coling 2008 Organizing Committee.

Maarten De Raedt, Fréderic Godin, Thomas Demeester, and Chris Develder. 2023. IDAS: Intent discovery with abstractive summarization. In *Proceedings of the 5th Workshop on NLP for Conversational AI (NLP4ConvAI 2023)*, pages 71–88, Toronto, Canada. Association for Computational Linguistics.

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.

Thomas Dopierre, Christophe Gravier, and Wilfried Logerais. 2021. PROTAUGMENT: Unsupervised diverse short-texts paraphrasing for intent detection meta-learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2454–2466, Online. Association for Computational Linguistics.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2023. MASSIVE: A 1M-example multilingual natural language understanding dataset with 51 typologically-diverse languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4277–4302, Toronto, Canada. Association for Computational Linguistics.

Lingyu Gao, Debanjan Ghosh, and Kevin Gimpel. 2023. The benefits of label-description training for zero-shot text classification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13823–13844, Singapore. Association for Computational Linguistics.

Spyros Gidaris and Nikos Komodakis. 2018. Dynamic few-shot visual learning without forgetting. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4367–4375.

Shai Gretz, Alon Halfon, Ilya Shnayderman, Orith Toledo-Ronen, Artem Spector, Lena Dankin, Yannis Katsis, Ofir Arviv, Yoav Katz, Noam Slonim, and Liat Ein-Dor. 2023. Zero-shot topical text classification with LLMs - an experimental study. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9647–9676, Singapore. Association for Computational Linguistics.

Milan Gritta, Ruoyu Hu, and Ignacio Iacobacci. 2022. CrossAligner & co: Zero-shot transfer methods for task-oriented cross-lingual natural language understanding. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 4048–4061, Dublin, Ireland. Association for Computational Linguistics.

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

Shailza Jolly, Tobias Falke, Caglar Tirkaz, and Daniil Sorokin. 2020. Data-efficient paraphrase generation to bootstrap intent classification and slot labeling for new features in task-oriented dialog systems. In *Proceedings of the 28th International Conference on Computational Linguistics: Industry Track*, pages 10–20, Online. International Committee on Computational Linguistics.

Ashutosh Kumar, Satwik Bhattamishra, Manik Bhandari, and Partha Talukdar. 2019. Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation. 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 3609–3619, Minneapolis, Minnesota. Association for Computational Linguistics.

Dmitry Lamanov, Pavel Burnyshev, Katya Artemova, Valentin Malykh, Andrey Bout, and Irina Piontkovskaya. 2022. Template-based approach to zero-shot intent recognition. In *Proceedings of the 15th International Conference on Natural Language Generation*, pages 15–28, Waterville, Maine, USA and virtual meeting. Association for Computational Linguistics.

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.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023a. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*.

Yuezhang Li, Ronghuo Zheng, Tian Tian, Zhiting Hu, Rahul Iyer, and Katia Sycara. 2016. Joint embedding of hierarchical categories and entities for concept categorization and dataless classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2678–2688, Osaka, Japan. The COLING 2016 Organizing Committee.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023b. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Han Liu, Xiaotong Zhang, Lu Fan, Xuandi Fu, Qimai Li, Xiao-Ming Wu, and Albert Y.S. Lam. 2019. Reconstructing capsule networks for zero-shot intent classification. 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 4799–4809, Hong Kong, China. Association for Computational Linguistics.

Han Liu, Siyang Zhao, Xiaotong Zhang, Feng Zhang, Junjie Sun, Hong Yu, and Xianchao Zhang. 2022. A simple meta-learning paradigm for zero-shot intent classification with mixture attention mechanism. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 2047–2052, New York, NY, USA. Association for Computing Machinery.

Samuel Louvan and Bernardo Magnini. 2020. Recent neural methods on slot filling and intent classification for task-oriented dialogue systems: A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 480–496, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Aaron Mueller, Jason Krone, Salvatore Romeo, Saab Mansour, Elman Mansimov, Yi Zhang, and Dan Roth. 2022. Label semantic aware pre-training for few-shot text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8318–8334, Dublin, Ireland. Association for Computational Linguistics.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. MTEB: Massive text embedding benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.

Thomas Müller, Guillermo Pérez-Torró, and Marc Franco-Salvador. 2022. Few-shot learning with Siamese networks and label tuning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8532–8545, Dublin, Ireland. Association for Computational Linguistics.

Guoshun Nan, Jiaqi Zeng, Rui Qiao, Zhijiang Guo, and Wei Lu. 2021. Uncovering main causalities for long-tailed information extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9683–9695, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, Johannes Heidecke, Pranav Shyam, Boris Power, Tyna Eloundou Nekoul, Girish Sastry, Gretchen Krueger, David Schnurr, Felipe Petroski Such, Kenny Hsu, Madeleine Thompson, Tabarak Khan, Toki Sherbakov, Joanne Jang, Peter Welinder, and Lilian Weng. 2022. Text and code embeddings by contrastive pre-training. *arXiv preprint arXiv:2201.10005*.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, and Yinfei Yang. 2022. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9844–9855, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Artem Popov, Victor Bulatov, Darya Polyudova, and Eugenia Veselova. 2019. Unsupervised dialogue intent detection via hierarchical topic model. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 932–938, Varna, Bulgaria. INCOMA Ltd.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381, Florence, Italy. Association for Computational Linguistics.

Gaurav Sahu, Pau Rodriguez, Issam Laradji, Parmida Atighehchian, David Vazquez, and Dzmitry Bahdanau. 2022. Data augmentation for intent classification with off-the-shelf large language models. In *Proceedings of the 4th Workshop on NLP for Conversational AI*, pages 47–57, Dublin, Ireland. Association for Computational Linguistics.

Sebastian Schuster and Christopher D. Manning. 2016. Enhanced English Universal Dependencies: An improved representation for natural language understanding tasks. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2371–2378, Portorož, Slovenia. European Language Resources Association (ELRA).

Yangqiu Song and Dan Roth. 2014. On dataless hierarchical text classification. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, page 1579–1585. AAAI Press.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2023. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1102–1121, Toronto, Canada. Association for Computational Linguistics.

Yixuan Su, Lei Shu, Elman Mansimov, Arshit Gupta, Deng Cai, Yi-An Lai, and Yi Zhang. 2022. Multi-task pre-training for plug-and-play task-oriented dialogue system. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4661–4676, Dublin, Ireland. Association for Computational Linguistics.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*.

Mina Valizadeh and Natalie Parde. 2022. The AI doctor is in: A survey of task-oriented dialogue systems for healthcare applications. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6638–6660, Dublin, Ireland. Association for Computational Linguistics.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Zhongyu Wei, Qianlong Liu, Baolin Peng, Huaixiao Tou, Ting Chen, Xuanjing Huang, Kam-fai Wong, and Xiangying Dai. 2018. Task-oriented dialogue system for automatic diagnosis. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 201–207, Melbourne, Australia. Association for Computational Linguistics.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449, Valencia, Spain. Association for Computational Linguistics.

Congying Xia, Chenwei Zhang, Xiaohui Yan, Yi Chang, and Philip Yu. 2018. Zero-shot user intent detection via capsule neural networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3090–3099, Brussels, Belgium. Association for Computational Linguistics.

Shitao Xiao, Zheng Liu, Yingxia Shao, and Zhao Cao. 2022. RetroMAE: Pre-training retrieval-oriented language models via masked auto-encoder. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 538–548, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighof. 2023. C-pack: Packaged resources to advance general chinese embedding. *arXiv preprint arXiv:2309.07597*.

Guangfeng Yan, Lu Fan, Qimai Li, Han Liu, Xiaotong Zhang, Xiao-Ming Wu, and Albert Y.S. Lam. 2020. Unknown intent detection using Gaussian mixture model with an application to zero-shot intent classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1050–1060, Online. Association for Computational Linguistics.

Zhao Yan, Nan Duan, Peng Chen, Ming Zhou, Jianshe Zhou, and Zhoujun Li. 2017. Building task-oriented dialogue systems for online shopping. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 4618–4625. AAAI Press.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. 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 3914–3923, Hong Kong, China. Association for Computational Linguistics.

Daochen Zha and Chenliang Li. 2019. Multi-label dataless text classification with topic modeling. *Knowl. Inf. Syst.*, 61(1):137–160.

Yiwen Zhang, Caixia Yuan, and Xiaojie Wang. 2021. Generalized zero-shot text classification via interclass relationship. In *2021 IEEE 7th International Conference on Cloud Computing and Intelligent Systems (CCIS)*, pages 413–417.

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 (Volume 1: Long Papers)*, pages 517–527, Dublin, Ireland. Association for Computational Linguistics.

A Utterance Paraphrasal

Table 7 contains an example template used to generate paraphrases for utterances from the CLINC dataset. Examples used in the template do not appear in the dataset and do not make explicit mentions of classes. We use `length_penalty=-1` to encourage shorter outputs, `repetition_penalty=1.2` and `num_beams=3`, we use default values for all other generation parameters.

We perform an additional ablation study over the choice of examples in the paraphrase generation template using 9 different examples across 3 configurations for each of SNIPS and MASSIVE datasets. We select these datasets specifically as we believe they differ sufficiently in number of intents and domains. Across 3 ablation configurations and the original paraphrasing setup, we obtain an overall score (mean of accuracy and macro-f1) of 92.66 $\pm$ 0.19% for SNIPS and 65.48 $\pm$ 0.18% for MASSIVE. As the standard deviation is low in both instances, we conclude that the choice of examples in the paraphrase generation prompt has little impact on the final performance through our setup.

Prompt
Given an utterance, describe what the user is asking. sentence: "set an alarm for every weekday at 7 am" description: user is asking to set an alarm for every weekday at 7am sentence: "can you show me the step-by-step instructions to bake chocolate chip cookies" description: user is asking for recipe for chocolate chip cookies sentence: "could you please tell me what time it is now" description: user is asking for the current time sentence: "{}" description:

Table 7: Example template used to generate user utterance paraphrases from the CLINC dataset.

B Example Masking Procedure

Given an user utterance "i want to watch animated movies at Showcase Cinemas", we first perform dependency parsing to identify utterance objects that can be masked. Figure 2 shows an illustration of the resulting parsed dependency relations. Following the approach outlined in Section 3.3.3, we mask out nodes with any of {`dobj, pobj, ccomp`} relations, namely "animated movies" and "Showcase Cinemas" to produce the resulting masked representation "i want to watch [MASK] at [MASK]".

C Details of selected models

Basic model specifications are shown in Table 8.

Model	s	d_h	l	$\mu_{\textbf{MTEB}}$
InstructOR$_{Large}$	1.34	768	512	61.59
E5-v2$_{Base}$	0.44	768	512	61.50
E5-v2$_{Large}$	1.34	1024	512	62.25
Multilingual-E5$_{Large}$	2.24	1024	514	61.50
E5$_{Large}$	1.34	1024	512	61.42
GTE$_{Small}$	0.07	384	512	61.36
GTE$_{Base}$	0.22	768	512	62.39
GTE$_{Large}$	0.67	1024	512	63.13
BGE$_{Small}$	0.13	384	512	62.17
BGE$_{Base}$	0.44	768	512	63.55
BGE$_{Large}$	1.34	1024	512	64.23
OpenAI-Ada-002	-	1536	8191	60.99

Table 8: Specifications of selected models grouped by training method. Column s shows model size (GB), d_h embedding dimensions, l maximum sequence length and $\mu_{\textbf{MTEB}}$ averaged performance on MTEB benchmark.

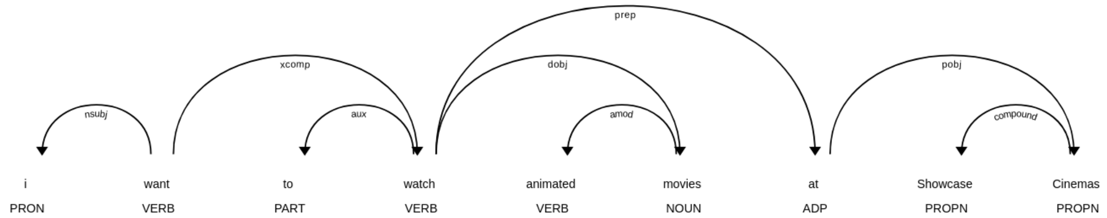

Figure 2: Example dependency parse tree from the SNIPS dataset.

InstructOR (Su et al., 2023) embeds the utterance with a task description, allowing for task-specific conditioning at inference time, with good performance on unseen domains. Trained on 330 datasets using a contrastive learning objective (Ni et al., 2022). This family of models is initialised from GTR (Ni et al., 2022) models, which are in-turn initialised from T5 (Raffel et al., 2020) models.

E5 (Wang et al., 2022) performs unsupervised pretraining on the model on $\sim$270M text pairs using an InfoNCE (van den Oord et al., 2019) objective with other utterances within the batch acting as negative examples, followed by supervised fine-tuning on 3 datasets. We select the *Base* and *Large* variants, initialised from *bert-base-uncased* and *bert-large-uncased-whole-word-masking* respectively.

GTE (Li et al., 2023b) pretrains the model on $\sim$800M text pairs and fine-tunes using 33 datasets. The contrastive learning objective used in this work considers, for each query-document pair (q_i, d_i) in a batch, the pairwise relation to the remaining examples $\{(q_j, d_j)\}_{j \neq i}$. The embedding similarities $s(q_i, d_j)$, $s(q_i, q_j)$, $s(d_i, d_j)$ are added to the partition function, where $s(q, d)$ is the cosine similarity between two embeddings.

BGE The work (Xiao et al., 2023) initialised from BERT (Devlin et al., 2019) models and trained using RetroMAE (Xiao et al., 2022) whereby both the input sentence and sentence embeddings in an autoencoder setup are randomly masked during MLM training. The authors use `[CLS]` token embeddings as the sentence representation. Our experimentation showed a slight improvement when using averaged token embeddings (Mean performance $+0.82\%$ *Tokenized-labels*, $+1.06\%$ *Class-description*).

D Full Results

See Table 9 for individual accuracy and macro-f1 scores by task and model.

E Further Ablations

We conduct further ablation studies using `bge-small-en-v1.5` (Table 10) and `gte-large` (Table 11) models to verify the findings of our main ablation study conducted on `bge-large-en-v1.5` (Table 4). We note that similar trends are observed with the different models, in that our proposed setup utilising a combination of the original utterance embedding with paraphrase embedding and masked utterance embedding using entity overlaps produced consistently higher scores.

F Description Paraphrasing

To produce paraphrases of intent descriptions, we leverage a `stablelm-2-1_6b-chat` model in a similar setup to our inference-time utterance paraphrasal. We increase temperature value from 0.5 to 4.1 in increments of 0.2, producing a paraphrase for each value. We then filter the generated set of descriptions for duplicates and enforce our *Label Preservation* and *Format Consistency* constraints, resulting in an average of 3.94 paraphrases per intent in addition to the original manually produced intent description. Each paraphrase has an average Levenshtein distance of 4.61 to the manual intent description. We replace half of all intent descriptions for each dataset with randomly sampled paraphrases, we produce 200 such combinations and repeat our experiments. Table 12 shows examples of paraphrased intent deescriptions for each dataset.

G t-SNE Visualisation

Due to the challenge to readability posed by the large number of intents in the CLINC dataset, instead sample the 15 top-performing (100% accuracy) and lowest-performing (24.47% accuracy) intent classes for illustration, with the results shown in Figures 1c and 1d respectively.

	Model	ATIS			SNIPS			CLINC			MASSIVE		
		Acc	F1	Mean	Acc	F1	Mean	Acc	F1	Mean	Acc	F1	Mean
Baselines	ICR	35.54	34.54	35.04	-	-	-	-	-	-	-	-	-
	SEG	-	-	-	69.61	69.31	69.46	-	-	-	-	-	-
	ML-SEG	-	-	-	77.08	75.97	76.53	-	-	-	-	-	-
	TIR$_{Orig}$	-	-	-	-	-	-	63.90	73.10	68.50	-	-	-
	TIR$_{Syn}$	-	-	-	-	-	-	58.00	61.30	59.65	-	-	-
	CosT	46.04	45.21	45.62	47.73	62.84	55.28	62.73	70.28	66.50	-	-	-
	LTA	66.09	**55.02**	60.55	90.09	84.22	87.16	73.18	75.74	74.46	-	-	-
	TTC$_D$	-	-	-	-	-	-	-	54.73	-	-	54.22	-
	Baselines	66.09	**55.02**	60.55	90.09	84.22	87.16	73.18	75.74	74.46	-	54.22	-
Tokenized Intent Labels	Instr.$_{Large}$	12.41	25.03	18.72	82.71	82.07	82.39	64.50	61.02	62.76	51.86	47.62	49.74
	E5-v2$_{Base}$	13.20	27.58	20.39	77.30	76.96	77.13	65.33	62.40	63.87	49.91	45.97	47.94
	E5-v2$_{Large}$	14.67	38.61	26.64	70.83	69.15	69.99	61.56	59.24	60.40	50.88	46.83	48.85
	mE5$_{Large}$	16.41	28.53	22.47	59.90	58.80	59.35	59.13	55.56	57.34	47.63	44.34	45.98
	E5$_{Large}$	44.71	36.43	40.57	75.68	73.21	74.44	70.27	67.96	69.11	51.30	49.78	50.54
	Ada-002	21.88	30.09	25.98	83.32	82.19	82.75	68.25	65.70	66.97	51.50	47.90	49.70
	GTE$_{Small}$	14.28	27.21	20.75	74.94	73.04	73.99	69.38	67.55	68.47	55.78	51.90	53.84
	GTE$_{Base}$	68.99	42.34	55.66	82.37	81.14	81.75	71.56	69.74	70.65	55.15	51.44	53.30
	GTE$_{Large}$	45.14	34.42	39.78	80.13	78.60	79.36	70.44	68.64	69.54	52.88	49.08	50.98
	BGE$_{Small}$	11.40	27.60	19.50	79.20	76.81	78.00	71.67	69.89	70.78	59.21	52.43	55.82
	BGE$_{Base}$	52.15	39.34	45.74	77.73	75.88	76.81	73.85	72.24	73.05	60.55	55.89	58.22
	BGE$_{Large}$	48.24	40.11	44.17	80.60	78.74	79.67	74.05	72.45	73.25	58.19	54.53	56.36
Intent Label Descriptions	Instr.$_{Large}$	41.24	43.12	42.18	85.85	85.35	85.60	77.95	76.55	77.25	57.95	55.52	56.73
	E5-v2$_{Base}$	64.84	40.04	52.44	87.75	87.23	87.49	72.15	69.68	70.92	55.57	53.73	54.65
	E5-v2$_{Large}$	62.33	41.98	52.16	87.84	86.77	87.31	72.39	70.59	71.49	57.30	55.65	56.48
	mE5$_{Large}$	75.85	45.16	60.51	84.64	83.11	83.88	73.09	71.39	72.24	60.09	56.67	58.38
	E5$_{Large}$	63.60	41.52	52.56	89.00	88.83	88.92	75.50	74.25	74.88	58.00	56.32	57.16
	Ada-002	58.97	43.71	51.34	89.71	89.28	89.50	78.75	76.86	77.81	59.49	58.03	58.76
	GTE$_{Small}$	66.62	42.80	54.71	84.62	84.22	84.42	71.19	69.22	70.20	55.18	51.86	53.52
	GTE$_{Base}$	63.21	41.99	52.60	86.60	86.22	86.41	75.90	74.30	75.10	56.47	54.62	55.55
	GTE$_{Large}$	66.91	44.79	55.85	86.65	86.01	86.33	76.62	75.04	75.83	59.27	57.85	58.56
	BGE$_{Small}$	55.69	39.99	47.84	86.01	85.01	85.51	73.04	71.01	72.03	57.31	54.27	55.79
	BGE$_{Base}$	53.14	44.37	48.76	88.66	87.98	88.32	78.38	76.85	77.61	60.91	58.92	59.91
	BGE$_{Large}$	62.07	47.70	54.88	89.58	89.01	89.30	79.70	78.46	79.08	<u>63.29</u>	<u>62.88</u>	<u>63.09</u>
+ Paraphrase and Masking	Instr.$_{Large}$	52.03	46.11	49.07	90.22	89.49	89.86	<u>80.82</u>	<u>79.51</u>	<u>80.17</u>	61.54	59.79	60.66
	E5-v2$_{Base}$	<u>78.39</u>	43.47	60.93	90.33	89.72	90.03	75.80	74.31	75.06	59.48	57.81	58.65
	E5-v2$_{Large}$	52.10	44.02	48.06	86.88	84.24	85.56	75.15	74.22	74.69	60.02	58.27	59.15
	mE5$_{Large}$	77.50	37.93	57.72	85.09	81.62	83.36	75.68	74.31	75.00	61.04	57.67	59.35
	E5$_{Large}$	65.37	42.19	53.78	91.96	<u>91.89</u>	<u>91.92</u>	76.40	76.13	76.27	61.01	59.17	60.09
	Ada-002	67.81	46.22	57.02	90.88	90.14	90.51	80.50	78.97	79.73	62.30	59.92	61.11
	GTE$_{Small}$	68.03	38.94	53.48	88.46	87.75	88.11	72.05	70.95	71.50	60.04	57.53	58.78
	GTE$_{Base}$	**80.50**	47.91	**64.20**	86.68	85.07	85.88	76.16	75.33	75.75	60.14	58.41	59.27
	GTE$_{Large}$	71.27	50.00	60.63	<u>92.00</u>	91.40	91.70	79.46	78.31	78.89	62.61	61.63	62.12
	BGE$_{Small}$	62.12	46.20	54.16	91.07	90.45	90.76	75.81	74.27	75.04	61.52	59.11	60.31
	BGE$_{Base}$	67.91	49.46	58.69	<u>92.00</u>	91.63	91.81	80.34	79.25	79.80	63.09	61.98	62.53
	BGE$_{Large}$	69.57	52.51	<u>61.04</u>	**92.81**	**92.33**	**92.57**	**81.95**	**81.09**	**81.52**	**65.49**	**65.76**	**65.62**

Table 9: Performance of baseline and selected models on 4 intent classification tasks. We report accuracy, macro-f1 score and the mean of both for each dataset. For each metric, **bold** denotes highest score, <u>underline</u> denotes second-highest

Setup	ATIS	SNIPS	CLINC	MASSIVE	Overall
embeds only	47.84	85.51	72.02	55.79	65.29
pp only	**55.57**	84.73	71.18	59.14	67.65
masked only	21.77	71.66	29.94	26.66	37.51
embeds + pp	52.87	86.83	**75.56**	<u>60.12</u>	68.85
embeds + masked	44.11	90.53	67.12	54.01	63.94
pp + masked	52.44	<u>91.16</u>	68.17	57.95	67.43
embeds + pp + masked	<u>54.16</u>	**91.19**	74.47	59.82	<u>69.91</u>
(overlap) embeds + masked	44.11	90.69	69.39	55.35	64.89
(overlap) pp + masked	52.44	90.68	69.41	58.32	67.71
(overlap) embeds + pp + masked	<u>54.16</u>	90.76	<u>75.04</u>	**60.23**	**70.05**

Table 10: Ablations on 4 intent classification datasets using a `bge-small-en-v1.5` model. **Overall** denotes the mean of accuracy and macro-f1 scores across all datasets.

Setup	ATIS	SNIPS	CLINC	MASSIVE	Overall
embeds only	55.85	86.33	75.83	58.56	69.14
pp only	51.39	83.93	75.87	60.49	67.92
masked only	35.15	75.00	35.71	31.45	44.33
embeds + pp	55.26	86.39	<u>78.86</u>	**62.29**	70.70
embeds + masked	**61.38**	**92.34**	72.92	57.10	70.94
pp + masked	59.17	91.69	73.21	59.86	70.98
embeds + pp + masked	60.64	91.89	78.64	61.97	<u>73.29</u>
(overlap) embeds + masked	**61.38**	<u>92.31</u>	74.41	57.91	71.50
(overlap) pp + masked	59.17	91.42	74.33	60.06	71.25
(overlap) embeds + pp + masked	60.64	91.70	**78.89**	<u>62.14</u>	**73.34**

Table 11: Ablations on 4 intent classification datasets using a `gte-large` model. **Overall** denotes the mean of accuracy and macro-f1 scores across all datasets.

H Embedding Similarities Analysis

We perform additional analysis on the mean embedding similarity of sentences within the same intent class (*in*-class) and of different intents (*out*-class). For a set of intent classes $\mathcal{C}$ and utterances $\mathcal{U}$, we calculate the mean *in*-class similarity $\mathbf{s}_{in}$ and *out*-class similarity $\mathbf{s}_{out}$ as

$$\mathbf{s}_{in} = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \sum_{u_i \in \mathcal{U}_c} \sum_{u_j \in \mathcal{U}_c \setminus \{u_i\}} \frac{s(\mathbf{h}(u_i), \mathbf{h}(u_j))}{n_c(n_c - 1)}$$

$$\mathbf{s}_{out} = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \sum_{u_i \in \mathcal{U}_c} \sum_{u_j \in \mathcal{U}_{c'}} \frac{s(\mathbf{h}(u_i), \mathbf{h}(u_j))}{n_c n_{c'}}$$

where $\mathcal{U}_c$ and $\mathcal{U}_{c'}$ denotes the set of utterances belonging to class c and all classes other than c' respectively, n_c is the number of utterances in set $\mathcal{U}_c$. The mean *in*-class and *out*-class similarity scores are shown per dataset (Table 13). From a basic correlation analysis of the mean embedding similarity against a number of metrics, we note for model performance on the MTEB benchmark there exists a strong positive correlation to the difference Δ_s between *in*-class and *out*-class examples (Pearson $r = 0.72$, $p < 0.01$) as well as $\%\Delta_s$ (Pearson $r = 0.73$, $p < 0.01$), and there exists a strong negative correlation to the mean *out*-class similarity $\mu_{s_{out}}$ (Pearson $r = -0.72$, $p < 0.01$).

I Synthetic Examples

We compare additionally against synthetic utterance generated for each intent class. We leverage `gpt-3.5-turbo` (OpenAI, 2023) for this purpose, by including the tokenized intent labels and label description within the prompt to generate a set $\mathcal{S}$ of questions or commands fitting said intent i.e. "*Given a category* `tokenized_intent` *and the description* `description`*, Please generate* `n` *different example sentences of users asking questions or making commands that fit the given category.*". At inference time, we sample k synthetic

Intent	Description	Paraphrase
`abbreviation`	user is asking what an abbreviation stands for or mean	"user is asking for a definition or explanation of an abbreviation" "user wants clarification on an abbreviation meaning" "user is asking about the meaning of an abbreviation"
`aircraft`	user is asking about an aircraft	"user is asking about an aircraft ticket or booking details" "user wants to know about an aircraft" "user wants information about an aircraft"
`airfare`	user is asking about fares, costs or airfares	"user wants to know airfare prices" "user wants to know about airfare prices"
`AddToPlaylist`	user wants to add a song to a playlist	"user wants to include a song in their playlist" "user wants to incorporate a song into their music collection" "user wants to add a song to their playlist"
`RateBook`	user wants the rating of/to rate a book	"user wants to give an opinion on a book" "user wants to leave a rating for a book" "user wants to leave a review on/ rate the book"
`SearchScreeningEvent`	user wants to know when a movie is on/screening time of a movie	"user wants movie screening information" "user wants to know movie screening schedule" "user wants to know movie screening time"
`accept_reservations`	user wants to know if a location accept reservations	"user wants to check if the place allows reservations" "user wants to check if a place allows reservations" "user wants to check location reservations"
`alarm`	user wants to set or get an alarm	"user wants a time alarm" "user wants to set a reminder or schedule an alarm" "user wants to set an alarm clock"
`calendar`	user wants to know about events from their calendar	"user is asking for event details from their calendar" "user wants to see their calendar for upcoming events" "user wants to check events in their calendar"
`email_query`	user is asking about email	"user wants to know how to send an email" "user wants to know how to use email effectively" "user wants an email response or clarification"
`general_greet`	user is saying a greeting	"user wants to talk or greet someone" "user wants to greet or say hello" "user wants to greet you or acknowledge your presence"
`news_query`	user is asking about the news	"user wants to learn about the latest news" "user wants to know the latest news" "user wants news update or clarification"

Table 12: Intents, descriptions and example paraphrases from all 4 intent classification datasets.

Dataset	$\mu_{s_{in}}$	$\sigma_{s_{in}}$	$\mu_{s_{out}}$	$\sigma_{s_{out}}$	Δ_s	$\%\Delta_s$
ATIS	0.80	0.06	0.77	0.06	0.03	3.86
SNIPS	0.76	0.04	0.69	0.05	0.07	8.65
CLINC	0.83	0.05	0.68	0.05	0.15	17.86
MASSIVE	0.80	0.05	0.69	0.05	0.11	13.73

Table 13: Mean embedding similarity of sentences within the same class (*in*) and different classes (*out*). Δ_s denotes the average difference between *in*-class and *out*-class, $\%\Delta_s$ denotes the percentage average difference of similarity.

k	Metric	ATIS		SNIPS		CLINC	
		μ	σ	μ	σ	μ	σ
$k=1$	Mean	23.59	8.42	71.37	5.51	53.87	5.42
	Δ_{Label}	-6.15	-4.23	-4.94	-1.02	-13.31	0.37
	Δ_{Desc}	-24.08	4.38	-15.54	2.57	-20.60	2.48
$k=3$	Mean	28.63	7.41	77.27	4.16	64.65	3.21
	Δ_{Label}	-1.10	-5.23	0.96	-2.37	-2.53	-1.84
	Δ_{Desc}	-19.03	3.37	-9.64	1.22	-9.82	0.27
$k=5$	Mean	30.05	6.74	78.54	3.98	67.29	2.81
	Δ_{Label}	0.31	-5.90	2.24	-2.55	0.11	-2.23
	Δ_{Desc}	-17.62	2.70	-8.36	1.04	-7.18	-0.13
$k=10$	Mean	30.80	5.33	79.63	3.57	69.24	2.48
	Δ_{Label}	1.06	-7.31	3.32	-2.96	2.06	-2.57
	Δ_{Desc}	-16.87	1.29	-7.28	0.63	-5.23	-0.46
$k=15$	Mean	31.12	5.15	80.06	3.46	69.99	2.50
	Δ_{Label}	1.38	-7.49	3.75	-3.07	2.80	-2.55
	Δ_{Desc}	-16.55	1.12	-6.85	0.52	-4.49	-0.44

Table 14: Averaged mean of accuracy and macro-f1 scores experiments conducted across 20 samples and 12 models using k number of synthetic examples per intent class. Δ_{Label} and Δ_{Desc} are differences to the averaged performance of methods using tokenized labels and intent descriptions respectively.

examples for c classes and make prediction $\hat{y}_i$ as follows:

$$\hat{y}_i = \arg\max_c \frac{\sum_m^k s(\mathbf{h}(u_i), \mathbf{h}(s_m^c))}{k}$$

where s_m^c denotes the m^{th} example utterance belonging to intent class $c \in \mathcal{C}$. Examples of synthetic utterances can be found in Appendix I. We report on the results separately in Section I.1 and the full results can be seen in Appendix J. We also consider synthetic examples generated using `gpt-4` but found the average performance to be lower on our task (Appendix K).

I.1 Results: Methods using Synthetic Data

We evaluate the efficacy of methods using synthetic examples by generating a set of $n = 20$ synthetic examples, from which we sample k to act as class prototypes, we repeat this procedure 20 times and compute the average performance across all samples. Table 14 shows averaged model performance across all 12 selected models and samples for $k = [1, 3, 5, 10, 15]$. For full results see Table 18 in Appendix J. We conducted additional experimentation with $k > 15$ but found further increasing k did not yield significant improvements in performance. We note our method using $k = 15$ synthetic examples outperforms tokenized labels on SNIPS (80.06 vs 76.30) and CLINC (69.99 vs 67.18) datasets, but underperforms slightly on the ATIS dataset (31.12 vs 31.70). Synthetic examples underperforms description-based methods by a considerable margin on all datasets, suggesting single intent label descriptions can be more powerful as class prototypes than synthetic instances. We note also the higher standard deviation σ in performance compared to the description-augmented method but lower compared to methods using tokenized labels.

I.2 Table of intents, descriptions and sampled synthetic examples generated using gpt-3.5-turbo

See Table 15 (ATIS), Table 16 (SNIPS) and Table 17 (CLINC).

J Full table of results for approach using synthetic examples generated using gpt-3.5-turbo

See Table 18.

K Table of averaged mean and standard deviation statistics for examples generated using gpt-4

See Table 19.

Intent	Description	Synthetic Examples
abbreviation	user is asking what an abbreviation stands for or mean	“what does eta stand for?" “can you tell me the meaning of atc?" “what is the abbreviation vfr referring to?"
aircraft	user is asking about an aircraft	“what is the maximum speed of this aircraft?" “can you provide me with the dimensions of the aircraft?" “how many passengers can this aircraft accommodate?"
airfare	user is asking about fares, costs or airfares	“what are the airfare options for a round-trip flight from new york to los angeles?" “can you provide me with the cost of a first-class airfare from london to paris?" “how much does it usually cost for a one-way airfare from tokyo to sydney?"
airline	user is asking about an airline/airlines	“which airline offers the most affordable tickets from los angeles to new york?" “can you recommend any airlines that provide extra legroom for tall passengers?" “what are the baggage restrictions for this airline?"
airport	user is asking about an airport/airports	“which airports in new york have direct flights to los angeles?" “can you provide me with information about the nearest airport to my current location?" “how long does it take to get from the city center to heathrow airport?"
capacity	user is asking about capacity (of an aircraft)	“what is the seating capacity of a boeing 747 aircraft?" “can you tell me the maximum passenger capacity of a airbus a380?" “what is the cargo capacity of a cessna 172 aircraft?"
cheapest	user is asking about the cheapest (fare)	“can you find me the cheapest flight from new york to los angeles?" “i need the cheapest airfare available for a one-way trip from london to barcelona." “what is the cheapest flight i can get from chicago to miami during the christmas holidays?"
city	user is asking about a city or place	“can you provide me with flight options to new york city?" “what are the popular attractions in san francisco?" “which airlines operate flights to tokyo?"
day_name	user is asking about a day (of the week)	“which day of the week is the best to book a flight?" “can you tell me the day of the week for my flight to new york?" “what is the departure day for the flight to london?"
distance	user is asking for the distance between places/locations	“what is the distance between new york and los angeles?" “calculate the distance from london to paris." “how far is it from sydney to melbourne?"
flight	user is asking about available flights	“what flights are available from new york city to los angeles tomorrow?" “can you please check if there are any direct flights from london to tokyo?" “i need to book a one-way flight from chicago to miami on the 15th of june."
flight_no	user is asking about a flight number	“what is the flight number for the flight from new york to london?" “can you provide me with the flight number for the 6:00 am departure to los angeles?" “i need to know the flight number for the red-eye flight to chicago."
flight_time	user is asking about departue time or schedule for a flight	“what is the flight time for the next available flight to new york?" “can you tell me the departure time for flight 123 to london?" “i need to know the schedule for flights leaving tomorrow morning."
ground_fare	user is asking about the ground fare at a destination	“what is the average ground fare in los angeles?" “can you provide information about ground fares in paris?" “how much should i expect to pay for ground transportation in london?"
ground_service	user is asking about ground service at a location	“what are the available ground services at this airport?" “can you provide me with information about ground services at the destination airport?" “is there wheelchair assistance available as part of the ground services?"
meal	user is asking about meals/catering	“what meal options are available for the flight?" “can i request a vegetarian meal for my flight?" “do you have any special meals for passengers with dietary restrictions?"
quantity	user is asking about the quantity/amount of something	“how many flight attendants are there on this flight?" “could you tell me the total weight of the luggage allowed per passenger?" “how many passengers are currently on board the plane?"
restriction	user is asking about restrictions	“can you please provide me with the baggage restrictions for my upcoming flight?" “what are the restrictions on carrying liquids in my hand luggage?" “are there any age restrictions for children traveling alone on your flights?"

Table 15: Intents, descriptions and synthetic examples for the ATIS dataset.

Intent	Description	Synthetic Examples
AddToPlaylist	user wants to add a song to a playlist	“hey, can you please add this new release to my workout playlist?" “add the latest hit by taylor swift to my party playlist, please." “can you include this classic rock track in my road trip playlist?"
BookRestaurant	user wants to book/make a reservation at a restaurant	“can you help me book a table at a fancy restaurant for this saturday?" “i would like to make a reservation for two at the most popular restaurant in town." “what is the best way to book a restaurant online?"
GetWeather	user wants to know about the weather	“what will be the weather like tomorrow?" “can you provide me with a detailed weather forecast for the next week?" “is it going to rain today?"
PlayMusic	user wants to play a song	“hey, playmusic! can you play ’shape of you’ by ed sheeran?" “playmusic, please play some soothing music to help me relax." “i’m in the mood for some throwback tunes. playmusic, can you play ’don’t stop believin’ by journey?"
RateBook	user wants the rating of/to rate a book	“can anyone recommend a ratebook website where I can find reviews and ratings for the latest bestsellers?" “what’s the highest-rated ratebook on the market right now? i want to make sure i’m picking something worthwhile." “i’d like some suggestions for popular ratebooks in the fantasy genre. any recommendations?"
SearchCreativeWork	user wants to find a creative work (book, song etc.)	“can you help me search for a creative work that is similar to harry potter?" “i’m looking for a book recommendation, search for a creative work with a thrilling mystery plot." “find me a song that has won multiple awards and has a catchy melody."
SearchScreeningEvent	user wants to know when a movie is on/screening time of a movie	“when is the next screening event for the movie avengers: endgame?" “what are the screening times for the romantic comedy crazy, stupid, love?" “can you tell me the showtimes for the movie joker in theaters nearby?"

Table 16: Intents, descriptions and synthetic examples for the SNIPS dataset.

Intent	Description	Synthetic Examples
timezone	user is asking about timezone	“can you please tell me the current timezone in new york city?" “what is the timezone difference between san francisco and tokyo?" “i need to know the exact timezone utc offset for london."
fun_fact	user wants to know a fun fact	“tell me a fun fact about the eiffel tower!" “i need a fun fact to impress my friends at dinner tonight." “give me a fun fact about dolphins, please."
date	user wants to know the current date	“what is today’s date?" “can you tell me the current date?" “i need to know the date, please."
what_can_i_ask_you	user wants to know what can be asked	“what kind of questions are appropriate to ask you?" “can you provide me with a list of topics that i can ask you about?" “i’m curious, what type of questions am i allowed to ask in this category?"
todo_list_update	user wants to update or change todo list	“how can i edit my current todo list?" “can you show me how to modify my existing tasks in the todo list?" “is there a way to change the priority of items on the todo list?"
bill_balance	user wants to know their bill balance	“what is my current bill balance?" “can you please provide the details of my bill balance?" “i need to know how much is due on my bill."
schedule_meeting	user wants to schedule meeting	“can you help me schedule a meeting for next week?" “i need assistance in setting up a meeting with our new client." “how do i go about scheduling a team meeting for tomorrow?"
routing	user wants to know about routing number	“what is a routing number and why is it important for banking?" “how can i find the routing number for my bank account?" “can you explain the specific purpose of a routing number in online transactions?"
food_last	user wants to know how long a food lasts	“how long can i safely keep cooked chicken in the refrigerator?" “what is the shelf life of fresh milk at room temperature?" “can you give me some tips on how to extend the life of avocados?"
bill_due	user wants to know when a bill is due	“hey, can you remind me when my electricity bill is due?" “what’s the due date for my credit card bill this month?" “i need to know when my phone bill is due. can you help me with that?"
time	user is asking for the time	“what is the current time?" “could you please tell me what time it is?" “do you have the time?"
freeze_account	user wants to freeze their account	“how can i freeze my account temporarily?" “i need to put a hold on my account, can you assist me?" “please freeze my account until further notice."
rollover_401k	user wants to know about 401k rollover	“how can i rollover my 401k into a new retirement account?" “can you explain the process of a 401k rollover to me?" “what are the benefits of doing a rollover with my 401k?"
travel_alert	user wants to know about travel alerts	“are there any current travel alerts that i should be aware of?" “notify me if there are any travel alerts for my upcoming destination." “can you provide me with the latest travel alerts for international travel?"
translate	user wants to translate	“can you translate this document from english to french?" “excuse me, i need assistance translating this menu into spanish." “how can i translate this phrase into italian?"

Table 17: Intents, descriptions and synthetic examples for 15 intents from the CLINC dataset.

	Model	ATIS			SNIPS			CLINC		
		Acc	F1	Mean	Acc	F1	Mean	Acc	F1	Mean
$n=1$	InstructOR$_{Large}$	32.77	23.99	28.38	72.60	69.26	70.93	56.94	53.71	55.32
	E5-v2$_{Base}$	27.01	19.30	23.16	70.28	66.52	68.40	50.05	47.21	48.63
	E5-v2$_{Large}$	29.50	19.12	24.31	68.09	64.41	66.25	47.24	44.54	45.89
	Multilingual-E5$_{Large}$	23.85	18.37	21.11	64.02	60.24	62.13	45.68	43.54	44.61
	E5$_{Large}$	28.57	20.22	24.40	69.35	66.13	67.74	54.44	51.38	52.91
	OpenAI-Ada-002	30.86	19.40	25.13	75.35	72.78	74.07	57.70	54.42	56.06
	GTE$_{Small}$	25.87	20.15	23.01	65.42	62.17	63.80	51.37	48.41	49.89
	GTE$_{Base}$	25.34	20.33	22.83	69.09	65.89	67.49	53.10	50.04	51.57
	GTE$_{Large}$	29.94	21.83	25.88	70.02	66.56	68.29	54.95	51.72	53.34
	BGE$_{Small}$	27.44	21.32	24.38	66.60	62.76	64.68	52.69	49.56	51.13
	BGE$_{Base}$	24.57	20.62	22.59	70.39	66.52	68.46	55.24	52.21	53.72
	BGE$_{Large}$	33.97	23.83	28.90	71.31	67.29	69.30	58.17	54.73	56.45
$n=3$	InstructOR$_{Large}$	39.20	29.25	34.22	76.71	72.39	74.55	67.88	64.84	66.36
	E5-v2$_{Base}$	35.75	26.97	31.36	76.25	71.56	73.90	63.52	60.63	62.08
	E5-v2$_{Large}$	40.41	27.85	34.13	75.68	70.98	73.33	62.35	59.47	60.91
	Multilingual-E5$_{Large}$	25.07	25.90	25.48	75.67	70.93	73.30	60.56	58.19	59.37
	E5$_{Large}$	37.33	29.64	33.48	74.57	70.24	72.40	67.18	64.25	65.72
	OpenAI-Ada-002	46.96	26.53	36.74	82.42	80.27	81.34	68.77	65.77	67.27
	GTE$_{Small}$	24.50	26.95	25.72	71.00	67.40	69.20	62.38	59.16	60.77
	GTE$_{Base}$	30.05	27.82	28.93	74.57	70.63	72.60	64.69	61.76	63.23
	GTE$_{Large}$	40.40	29.40	34.90	75.04	71.23	73.14	65.78	62.67	64.23
	BGE$_{Small}$	29.24	27.49	28.37	73.49	68.98	71.23	64.59	61.72	63.16
	BGE$_{Base}$	28.35	27.00	27.67	73.83	69.23	71.53	66.59	63.66	65.13
	BGE$_{Large}$	38.30	28.14	33.22	74.83	70.09	72.46	68.05	64.62	66.34
$n=5$	InstructOR$_{Large}$	41.77	32.86	37.31	78.36	74.08	76.22	70.30	67.51	68.90
	E5-v2$_{Base}$	34.49	28.76	31.63	78.53	73.47	76.00	66.75	63.94	65.34
	E5-v2$_{Large}$	36.82	29.53	33.17	78.02	73.66	75.84	65.70	62.76	64.23
	Multilingual-E5$_{Large}$	31.29	29.28	30.29	76.21	72.18	74.19	64.36	61.78	63.07
	E5$_{Large}$	37.24	32.79	35.01	76.04	71.20	73.62	69.63	66.62	68.13
	OpenAI-Ada-002	45.01	28.38	36.70	84.56	82.60	83.58	70.81	68.03	69.42
	GTE$_{Small}$	32.92	30.05	31.48	73.21	69.16	71.18	65.63	62.58	64.10
	GTE$_{Base}$	29.90	30.02	29.96	76.54	72.13	74.33	67.11	63.95	65.53
	GTE$_{Large}$	41.92	32.41	37.17	75.73	71.18	73.45	68.48	65.38	66.93
	BGE$_{Small}$	35.33	32.64	33.99	72.85	68.06	70.46	67.15	64.35	65.75
	BGE$_{Base}$	27.94	29.49	28.72	76.61	71.90	74.25	69.42	66.52	67.97
	BGE$_{Large}$	35.79	32.38	34.08	76.26	71.00	73.63	70.68	67.64	69.16
$n=10$	InstructOR$_{Large}$	47.38	33.77	40.58	80.58	76.50	78.54	72.37	69.68	71.03
	E5-v2$_{Base}$	37.04	32.17	34.60	80.31	74.92	77.61	69.59	66.86	68.23
	E5-v2$_{Large}$	46.80	32.53	39.66	79.11	74.31	76.71	68.65	65.70	67.17
	Multilingual-E5$_{Large}$	30.88	32.70	31.79	78.71	74.43	76.57	67.87	65.39	66.63
	E5$_{Large}$	41.44	34.74	38.09	77.83	73.35	75.59	72.42	69.62	71.02
	OpenAI-Ada-002	46.60	32.90	39.75	85.57	83.46	84.51	73.30	70.60	71.95
	GTE$_{Small}$	32.71	33.53	33.12	74.77	70.42	72.59	67.48	64.56	66.02
	GTE$_{Base}$	28.05	31.23	29.64	77.35	72.76	75.06	69.50	66.44	67.97
	GTE$_{Large}$	45.05	35.25	40.15	76.29	71.67	73.98	69.86	66.90	68.38
	BGE$_{Small}$	36.24	34.44	35.34	75.95	71.13	73.54	68.96	66.27	67.61
	BGE$_{Base}$	31.14	31.62	31.38	78.15	73.07	75.61	71.48	68.73	70.10
	BGE$_{Large}$	43.19	35.56	39.38	77.77	72.44	75.10	72.36	69.39	70.88
$n=15$	InstructOR$_{Large}$	40.59	35.40	37.99	80.57	75.75	78.16	73.10	70.54	71.82
	E5-v2$_{Base}$	42.17	34.44	38.31	80.25	74.65	77.45	70.18	67.50	68.84
	E5-v2$_{Large}$	47.71	33.67	40.69	79.86	74.66	77.26	69.70	66.69	68.19
	Multilingual-E5$_{Large}$	28.31	33.48	30.89	79.91	75.32	77.61	69.31	66.76	68.03
	E5$_{Large}$	42.42	36.31	39.36	78.02	73.00	75.51	73.13	70.26	71.69
	OpenAI-Ada-002	48.13	34.26	41.20	87.04	85.03	86.03	73.97	71.36	72.66
	GTE$_{Small}$	38.54	34.38	36.46	75.03	70.32	72.68	68.63	65.60	67.12
	GTE$_{Base}$	33.68	32.35	33.02	78.27	73.56	75.92	69.86	66.73	68.29
	GTE$_{Large}$	37.98	34.38	36.18	77.78	72.93	75.36	70.51	67.62	69.07
	BGE$_{Small}$	28.06	34.30	31.18	75.43	70.54	72.98	70.20	67.56	68.88
	BGE$_{Base}$	27.20	31.08	29.14	78.92	73.65	76.29	71.93	69.15	70.54
	BGE$_{Large}$	42.22	37.06	39.64	78.76	73.43	76.10	73.17	70.24	71.71

Table 18: Results per model using k synthetic examples averaged across 20 samples.

k	Metric	ATIS		SNIPS		CLINC	
		μ	σ	μ	σ	μ	σ
$k=1$	Mean	24.51	10.15	67.63	5.48	51.63	5.13
	Δ_{Label}	-7.19	-2.58	-8.68	-1.05	-15.56	0.08
	Δ_{Desc}	-27.38	6.37	-19.29	2.46	-22.92	2.12
$k=3$	Mean	31.19	8.61	73.25	4.49	63.71	2.76
	Δ_{Label}	-0.51	-4.11	-3.06	-2.04	-3.47	-2.29
	Δ_{Desc}	-20.70	4.84	-13.66	1.47	-10.83	-0.25
$k=5$	Mean	33.29	7.90	74.73	4.16	66.54	2.35
	Δ_{Label}	1.59	-4.82	-1.57	-2.37	-0.64	-2.70
	Δ_{Desc}	-18.60	4.13	-12.18	1.14	-8.00	-0.67
$k=10$	Mean	36.12	7.51	76.28	3.49	68.92	2.08
	Δ_{Label}	4.42	-5.21	-0.02	-3.04	1.73	-2.97
	Δ_{Desc}	-15.77	3.73	-10.63	0.48	-5.63	-0.94
$k=15$	Mean	36.17	7.13	76.78	3.75	69.74	1.93
	Δ_{Label}	4.47	-5.59	0.48	-2.78	2.55	-3.12
	Δ_{Desc}	-15.72	3.36	-10.13	0.73	-4.81	-1.09

Table 19: Averaged mean of accuracy and macro-f1 scores experiments conducted across 20 samples and 12 models using k number of synthetic examples per intent class generated using `gpt-4-1106-preview`. Δ_{Label} and Δ_{Desc} are differences to the averaged performance of methods using tokenized labels and intent descriptions respectively.

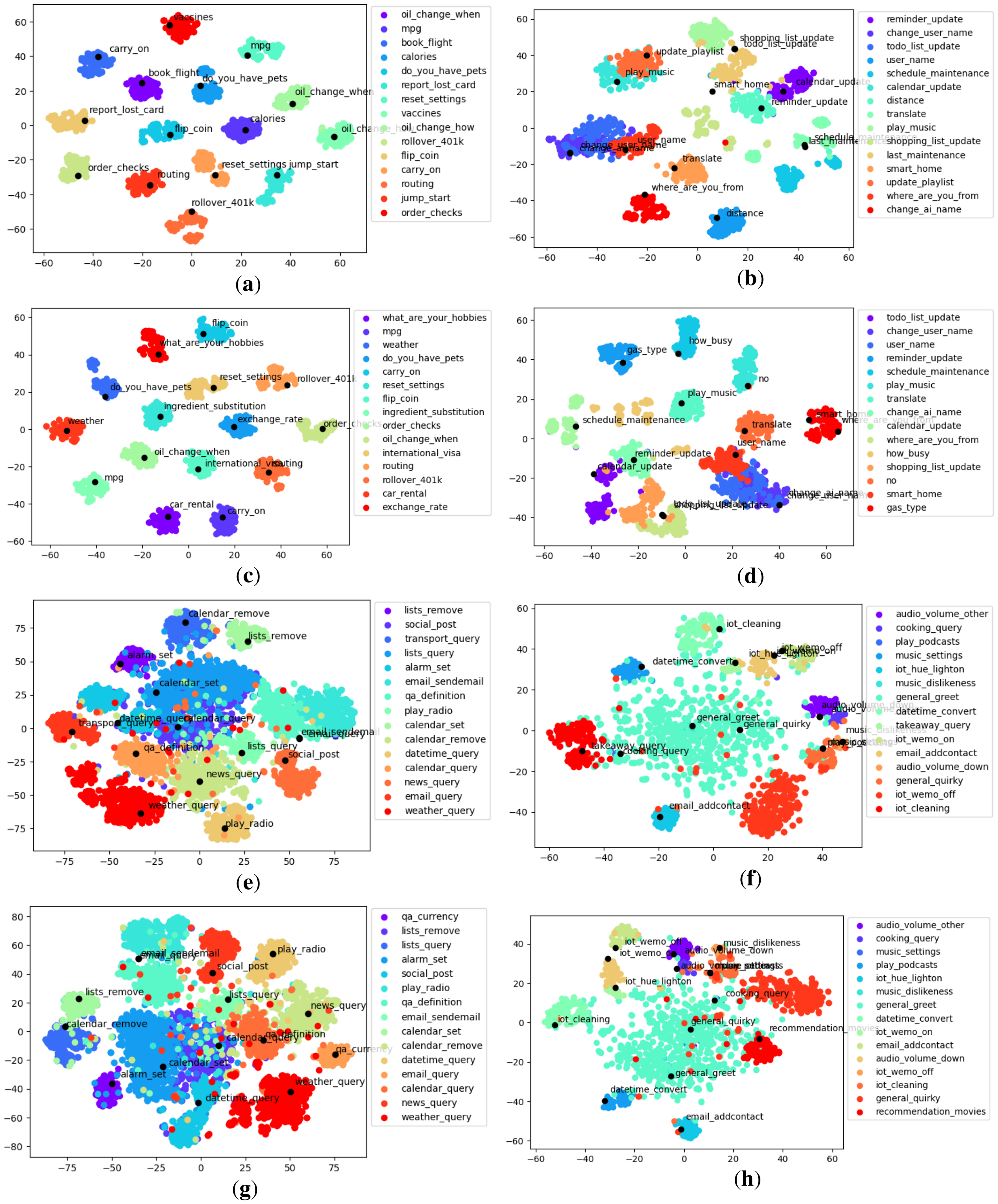

Figure 3: t-SNE (van der Maaten and Hinton, 2008) visualisation of embeddings for CLINC and MASSIVE datasets computed using BGE$_{Large}$, class label description embeddings are shown in black and labelled. (**Row 1**) Embeddings of top 15 and bottom 15 classes from CLINC, (**Row 2**) Embedding + Paraphrasing and Masking of top 15 and bottom 15 classes from CLINC, (**Row 3**) Embeddings for top 15 and bottom 15 classes from MASSIVE, (**Row 4**) Embedding + Paraphrasing and Masking of top 15 and bottom 15 classes from CLINC.