LANID: LLM-assisted New Intent Discovery

Lu Fan[1], Jiashu Pu[2], Rongsheng Zhang[2], and Xiao-Ming Wu[1]*
Department of Computing, The Hong Kong Polytechnic University, Hong Kong S.A.R.[1]
Fuxi AI Lab, NetEase Inc. China, Hang Zhou, China.[2]
cslfan@comp.polyu.edu.hk, Fuxi AI Lab, NetEase Inc.,
zhangrongsheng@corp.netease.com, xiao-ming.wu@polyu.edu.hk

Abstract

Data annotation is expensive in Task-Oriented Dialogue (TOD) systems. New Intent Discovery (NID) is a task aims to identify novel intents while retaining the ability to recognize known intents. It is essential for expanding the intent base of task-based dialogue systems. Previous works relying on external datasets are hardly extendable. Meanwhile, the effective ones are generally depends on the power of the Large Language Models (LLMs). To address the limitation of model extensibility and take advantages of LLMs for the NID task, we propose LANID, a framework that leverages LLM's zero-shot capability to enhance the performance of a smaller text encoder on the NID task. LANID employs KNN and DBSCAN algorithms to select appropriate pairs of utterances from the training set. The LLM is then asked to determine the relationships between them. The collected data are then used to construct finetuning task and the small text encoder is optimized with a triplet loss. Our experimental results demonstrate the efficacy of the proposed method on three distinct NID datasets, surpassing all strong baselines in both unsupervised and semi-supervised settings. Our code can be found in https://github.com/floatSDSDS/LANID.

Keywords: Out-of-distribution detection, large language models, performance evaluation, clustering

1. Introduction

In recent times, advancements in Large Language Models' (LLMs) zero-shot capabilities (Heck et al., 2023) suggest that task-oriented dialogue (TOD) systems may eventually be replaced by a universal model. Nevertheless, the current dependence on third-party LLMs still poses concerns regarding network communication and data privacy breaches. Thus, we contend that TOD systems still have a role to play. These systems rely on comprehending user input and precisely discerning their requirements while also consistently updating and maintaining intents for the Natural Language Understanding (NLU) Module.

Due to the high cost of manual annotation, previous work has proposed methods to automatically do New Intents Discovery (NID) from the utterances of conversational systems (Lang et al., 2022; Zhang et al., 2022; Manik et al., 2021). These works design novel new learning strategies and architectures for NID tasks. However, the latter two methods necessitate the use of external datasets and knowledge graphs, and it remains unclear whether they can be effectively employed in particular domains. Furthermore, the efficacy of these methods hinges on the potent representational prowess of the LLM. To improve the NID results, one potential solution is to improve the LLM itself. Currently, GPT4 is arguably the most potent LLM available (OpenAI, 2023), but fine-tuning it with domain datasets is not feasible due to its closed-source nature.

In order to employ the strongest LLM for the NID task, we design a framework that utilizes LLM's impressive zero-shot capability to aid a smaller text encoder in acquiring utterance representations and enhancing the in-domain NID outcomes.

The framework is named LANID — **L**LM-**A**ssisted **N**ew **I**ntent **D**iscovery. The LANID method comprises of several key phases. Initially, we utilize KNN and DBSCAN algorithms to select appropriate pairs of utterances from the training dataset. This selection process takes into account both local and global distributions to accurately reflect the overall distribution of the domain data. Subsequently, we employ LLM's zero-shot functionality to determine the relationships between the chosen utterance pairs. Once the relationship labels are obtained, we develop a triplet margin loss to guide the training of the small text encoder, aiming to refine its representation of the domain data. Through multiple iterations of these steps, we leverage the small text encoder to extract representations that enable us to perform the NID task through clustering. Our experimental design encompasses both unsupervised and semi-supervised settings, and we demonstrate the effectiveness of the LANID method on three distinct NID datasets, where it outperforms all strong baselines.

2. Related Works

The study of New Intent Discovery (NID) is an active research area with several types of approaches

*Corresponding author

LREC-COLING 2024, pages 10110–10116

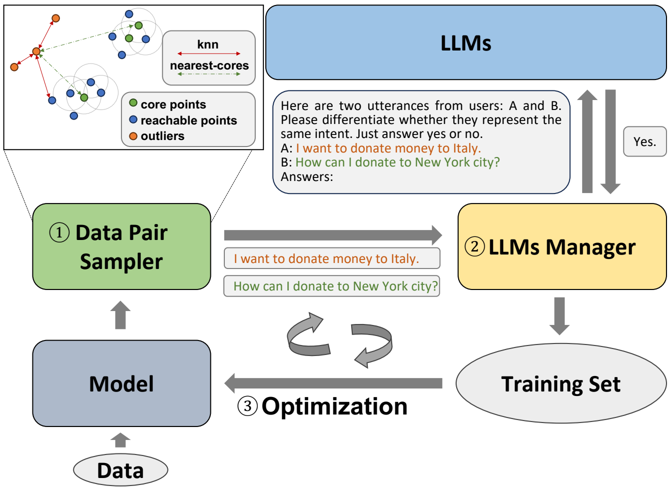

Figure 1: Illustration of the proposed LANID framework.

proposed. In the early stages of NID research, unsupervised clustering methods (Shi et al., 2018; Perkins and Yang, 2019; Chatterjee and Sengupta, 2020) were commonly explored. However, they cannot utilize the existing labeled data in the system and deviate from the practical situation.

To leverage known labels as well as discover unknown intent, a more proper way is to apply semi-supervised training scheme (Lin et al., 2020; Zhang et al., 2021a,b, 2022; Pu et al., 2022). However, these methods often rely on smaller semantic encoders, such as BERT (Devlin et al., 2018), which can only provide limited general knowledge for intent representation. In this paper, we leverage the powerful semantic understanding capabilities of large language model to generate auxiliary labels for contrastive training.

3. Problem Formulation

In practice, we often need to mine new intents from the mass of utterances in a TOD system, which can be built either from scratch or as an upgrade to an earlier system. To consider both scenarios, we follow the prior research (Zhang et al., 2022) and adopt unsupervised and semi-supervised evaluation settings. We denote the utterance, its intent label, the set of unseen intents, the set of seen intents, the training set, and the test set as x, y, $\mathcal{C}_u$, $\mathcal{C}_k$, D_{train}, and D_{test} respectively. In the unsupervised setting, D_{train} does not contain any labels, and we aim to group utterances from D_{test} with similar intent into a cluster, each cluster being a new intent (belongs to $\mathcal{C}_u$). While in the semi-supervised setting, D_{train} contains both labeled dataset $D_{labeled} = \{(x_i, y_i)|y_i \in \mathcal{C}_k\}$ and an unlabeled dataset $D_{unlabeled} = \{x_i|y \in \{\mathcal{C}_k, \mathcal{C}_u\}\}$, our goal is to discriminate existing intents from D_{test}, while mining for novel intents in the remaining utterances.

4. Method

Our approach is to use a text encoder to extract features from utterances and then do clustering to mine new intents. There are three main steps at training: 1) selecting the utterance pairs from D_{train} that represents local and global information 2) requesting LLM to determine the relationship between the utterance pairs 3) incorporating the output of the LLM into the triplet margin loss on which the parameters of the text encoder are updated. The above three steps are repeated until convergence. After that, we do clustering on D_{test} based on the learned representations. We summarize the process in Figure 1.

4.1. Selecting the Utterance Pairs

Using an off-the-shelf text encoder to extract utterance representations is suboptimal because the focus on mining new intentions varies across different domains. Therefore, we need to quickly adapt the text encoder to new domains. We propose to utilize the LLM's powerful zero-shot capability to determine the relationship between utterance pairs in the current domain, allowing us to adjust the text encoder's parameters. Selecting appropriate utterance pairs that accurately and comprehensively represent the data distribution in the new domain is critical. To this end, we chose to select utterance pairs from both local and global perspectives.

Selection based on K Nearest Neighbors. Starting locally, we first find those utterances that are close to each other (based on the original representation) and determine whether the distribution

Table 1: Hyper-parameter settings. MinPts refers to the minimum number of points within a specified radius (epsilon) that are required to form a dense region in DBSCAN.

	K	p	n_k	m	k_n	T	#Epoch	MinPts
Banking	50	0.1	2	5	2	3	10	4
Stackoverflow	50	0.05	2	8	2	2	10	4
M-CID	50	0.2	2	5	2	3	20	4

Table 2: Performance on unsupervised NID. For each dataset, the best results are marked in bold. LANID-Near only adopts KNN-based sampling strategy, LANID-DBSCAN adopts only DBSCAN sampling strategy, and LANID (combined) is a mixture of both strategies.

		Banking			StackOverflow			M-CID		
	Methods	**NMI**	**ARI**	**ACC**	**NMI**	**ARI**	**ACC**	**NMI**	**ARI**	**ACC**
unsupervised	SAE-KM	60.12	24.00	37.38	48.72	23.36	37.16	51.03	43.51	52.95
	SAE-DEC	62.92	25.68	39.35	61.32	21.17	57.09	50.69	44.52	53.07
	SAE-DCN	62.94	25.69	39.36	61.34	34.98	57.09	50.69	44.52	53.07
	MTP	77.25	47.80	59.12	61.35	45.77	61.90	70.53	45.76	64.76
	MTP-CLNN	82.15	57.68	66.88	75.20	63.13	79.20	80.03	67.39	79.94
	LANID-Near	83.44	58.28	66.75	79.56	66.67	83.40	80.80	69.86	81.38
	LANID-DBSCAN	83.21	58.02	65.78	81.25	72.86	85.30	80.41	68.10	79.08
	LANID	**84.12**	**60.40**	**70.58**	**81.25**	**72.96**	**86.60**	**82.64**	**71.36**	**82.52**

among them is reasonable. Specifically, we randomly sample $p\%$ utterances from D_{train}. Then, for each sampled utterance x_i, we search for its top-K nearest neighbors $\mathcal{N}_i^{Near}$ using the Euclidean distance, and we uniformly sample $n_k (n_k < |\mathcal{N}_i^{Near}|)$ utterances from $\mathcal{N}_i^{Near}$. We denote the nearest-neighbor set for x_i as $\mathcal{M}_i^{Near} = \{(x_i, x_j)|x_j \in \mathcal{N}_i^{Near}\}$, where $|\mathcal{M}_i^{Near}| = n_k$.

Selection based on Global Density. In general, it is difficult to divide a data set into exactly a few categories, and there will always be some outliers. Also, the distribution of semantics is usually not uniform, and there are high and low densities of different semantic clusters. We propose a DBSCAN-based (Ester et al., 1996) sampling approach to reflect the relationship between globally high and low-density regions of semantics. Concretely, we conduct DBSCAN clustering on D_{train} and obtain a set of core points $\mathbf{x}_c$ and a set of non-core points $\mathbf{x}_{nc}$. Then, we randomly sample a subset $\mathbf{x}_{nc}^{'}$ from $\mathbf{x}_{nc}$. For each utterance x_i in $\mathbf{x}_{nc}^{'}$, we search for its m nearest neighbors in $\mathbf{x}_c$, forming a global density set as $\mathcal{M}_i^{Den} = \{(x_i, x_j)|x_j \in \mathcal{N}_i^{Core}\}$, where $\mathcal{N}_i^{Core}$ is the set consisting of the nearest points to x_i in $\mathbf{x}_c$.

4.2. LLM Manager

The LLM manager is the other major module in LANID. It constructs prompts with sampled data and parse the responses from LLMs.

We construct promompts with three components (Pan et al., 2023), namely schema, regulations, and sentence input. The schema component aims to prompt LLMs to produce responses that meet our desired criteria. To identify an optimal schema for each dataset, several schemas were manually crafted and subsequently evaluated based on their performance on $D_{labeled}$. The regulations component constrains the format of LLM's responses. We chose to use the phrase "just answer yes or no" uniformly for simplicity. Thirdly, the sentence input component consists of utterance pairs that are sampled as detailed in Section 4.1. Finally, we predict $r(i, j) = 1$ for a data pair (i, j) if 'yes' is in the LMs' corresponding response otherwise $r(i, j) = 0$. The regulations component constrains the format of LM's responses. We chose to use the phrase "just answer yes or no" uniformly for simplicity.

4.3. Training and Optimization

To optimize the representation of the text encoder on the domain data, we collect pairs of positive samples from $\mathcal{M}_i^{Near}$ or/and $\mathcal{M}_i^{Core}$, with their relationships $r(i, j)$ determined by the LLM manager. For each positive sample pair $\{(x_i, x_j)\}$, we directly sample k_n utterances at random from D_{train} as negative samples to better represent the distribution of the whole dataset (we assume that the distribution of each dataset is not extreme). In this way, we form a dataset $D_f = (x_i, p_i, n_i)$ of triplets. Then, we finetune the model with a triplet margin loss defined as:

$$\begin{aligned}\mathcal{L}(x_i, &p_i, n_i) = max(\\ &d(x_i, p_i) - d(x_i, n_i) + margin, 0),\end{aligned} \tag{1}$$

where x_i here works as the anchor point, p_i and n_i is its positive and negative, respectively. $d(x_i, y_j) = \|x_i - y_j\|$ and the margin value is a hyperparameter that determines the minimum desired difference between $d(x_i, p_i)$ and $d(x_i, n_i)$.

As the training proceeds, the quality of the text encoder's representation improves, leading to en-

Table 3: Performance on semi-supervised NID with different known class ratio. For each dataset, the best results are marked in bold. Known Class Ratios (KCR) is defined as $\frac{|\mathcal{C}_k|}{(|\mathcal{C}_k|+\mathcal{C}_u)}$. We randomly sampled a 10% subset for each known class to form the $D_{labeled}$.

		Banking			StackOverflow			M-CID		
	Methods	**NMI**	**ARI**	**ACC**	**NMI**	**ARI**	**ACC**	**NMI**	**ARI**	**ACC**
	BERT-KCL	53.85	20.07	28.79	35.47	16.80	32.88	29.35	11.58	24.76
	DAC	69.85	37.16	49.67	53.97	36.46	53.96	49.83	27.21	43.72
	MTP	79.17	50.83	62.05	74.86	62.27	77.20	70.53	45.76	64.76
	MTP-CLNN	83.88	60.76	70.91	78.38	65.80	80.10	78.30	65.32	78.30
KCR-25%	LANID-Near	85.28	63.48	**72.47**	**80.83**	**65.86**	**83.30**	81.91	70.30	81.09
	LANID-DBSCAN	84.74	62.22	70.13	74.74	60.54	73.70	80.04	69.69	83.09
	LANID	**85.51**	**64.23**	71.40	79.55	63.23	81.80	**85.11**	**75.66**	**86.82**
		Banking			**StackOverflow**			**M-CID**		
	Methods	**NMI**	**ARI**	**ACC**	**NMI**	**ARI**	**ACC**	**NMI**	**ARI**	**ACC**
	BERT-KCL	62.86	30.16	40.81	57.63	41.90	56.58	42.48	22.83	38.11
	DAC	76.41	47.28	59.32	70.78	56.44	73.76	63.27	43.52	57.19
	MTP	82.12	56.43	67.34	76.58	65.55	82.50	70.53	45.76	64.76
	MTP-CLNN	86.42	66.66	74.97	81.41	**72.15**	**86.00**	79.34	66.18	78.80
KCR-50%	LANID-Near	**86.83**	**67.41**	**76.10**	81.62	64.32	81.30	81.20	69.54	81.95
	LANID-DBSCAN	85.62	64.35	72.44	81.19	65.75	81.40	79.16	67.85	80.80
	LANID	86.31	66.53	75.49	**82.07**	70.51	83.00	**81.58**	**70.66**	**82.81**
		Banking			**StackOverflow**			**M-CID**		
	Methods	**NMI**	**ARI**	**ACC**	**NMI**	**ARI**	**ACC**	**NMI**	**ARI**	**ACC**
	BERT-KCL	72.18	44.29	58.70	70.38	57.98	71.50	54.22	34.60	52.15
	DAC	79.99	54.57	65.87	75.31	60.02	78.84	71.41	54.22	69.11
	MTP	84.61	63.23	72.76	80.41	70.01	81.10	77.90	64.57	77.65
	MTP-CLNN	87.24	68.77	77.14	80.99	72.14	85.80	80.12	67.40	79.37
KCR-75%	LANID-Near	87.59	**70.13**	**78.51**	82.14	73.05	85.80	81.13	69.75	**83.09**
	LANID-DBSCAN	86.79	67.18	74.35	**83.74**	**76.45**	**88.30**	80.65	70.24	82.52
	LANID	**87.64**	68.89	76.56	82.80	74.33	87.50	**82.16**	**70.56**	82.23

hanced sampling outcomes. In practice, the procedure of choosing the utterance pairs and requesting the LLM manager recurs every T epochs, with the fine-tuning dataset D_f and the text encoder being incrementally updated during this iterative process.

5. Experiment

5.1. Experimental Details

Datasets. We evaluate LANID on three intent recognition benchmarks. BANKING (Casanueva et al., 2020) encompasses 13,083 utterances distributed across 77 intents in the banking domain. StackOverflow (Xu et al., 2015) comprises 20,000 queries collected from an online question-answering platform[1], categorized into 20 categories. M-CID (Arora et al., 2020) consists of 1,745 utterances associated with 16 intents specifically related to Covid-19 services.

Experimental Setup. Our proposed method is evaluated under both unsupervised and semi-supervised settings. We employed three clustering evaluation metrics, namely normalized mutual information (NMI), adjusted rand index (ARI) (Yeung and Ruzzo, 2001), and accuracy (ACC).

Baselines. We compare LANID with both unsupervised and semi-supervised NID SOTAs. Unsupervised NID SOTAs include SAE (Xie et al., 2016) series, MTP, and CLNN (Zhang et al., 2022). Semi-supervised baselines includes BERT-KCL (Hsu et al., 2019), DAC (Zhang et al., 2021b), MTP and CLNN (Zhang et al., 2022).

Implementation Details. We use default settings of CLNN (Zhang et al., 2022), and continue to train the model pretrained by MTP-CLNN as a post fine-tuning stage. It is also possible to conduct further training in other NID baselines. As for LLMs, we use *gpt-3.5-turbo*[2] model. The hyper-parameters of LANID are selected based on the performances on the validation set. The parameters are shown in Table 1.

5.2. Result Analysis

Table 2 and Table 3 summarizes the performance of LANID in comparison to unsupervised SOTAs and semi-supervised SOTAs on three intent recognition benchmarks. The results reveal several key observations. (1) LANID and its variants demonstrate impressive performance under the unsupervised learning settings, outperforming other baselines. This can be attributed to the proficient guidance provided by the LLM labeling, which effectively compensates for the absence of supervised signals. (2) LANID consistently outperforms the

[1]https://stackoverflow.com/

[2]https://platform.openai.com/docs/models/gpt-3-5

baselines in almost all cases, highlighting its efficacy. (3) Although the combination of two neighborhood sampling strategies works well, relying solely on the DBSCAN-based sampling strategy can sometimes hinder performance. This is due to the fact that the LLM is constraints to make binary judgments and retain only positive pairs. For many outliers, their nearest cores may not express the same intent, thereby reducing the size of D_f and leading to overfitting problems.

6. Conclusion

This paper presents a novel framework, LANID, which utilizes LLM for solving the NID problem. Rather than asking LLM to directly recognize new intents, our approach employs LLMs to extract relations among data and construct fine-tuning tasks accordingly. To improve data sampling efficiency, we propose two neighborhood-based sampling strategies for selective data pair sampling. Extensive experiments on three intent recognition benchmarks demonstrate the effectiveness of our proposed method.

Acknowledgments

We thank the anonymous reviewers for their valuable feedback. This research was partially supported by the grant of HK ITF ITS/359/21FP.

7. Bibliographical References

Alfred V. Aho and Jeffrey D. Ullman. 1972. *The Theory of Parsing, Translation and Compiling*, volume 1. Prentice-Hall, Englewood Cliffs, NJ.

American Psychological Association. 1983. *Publications Manual*. American Psychological Association, Washington, DC.

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853.

Galen Andrew and Jianfeng Gao. 2007. Scalable training of L_1-regularized log-linear models. In *Proceedings of the 24th International Conference on Machine Learning*, pages 33–40.

Abhinav Arora, Akshat Shrivastava, Mrinal Mohit, Lorena Sainz-Maza Lecanda, and Ahmed Aly. 2020. Cross-lingual transfer learning for intent detection of covid-19 utterances.

Vassileios Balntas, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. 2016. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *Bmvc*, volume 1, page 3.

BSI. 1973a. *Natural Fibre Twines*, 3rd edition. British Standards Institution, London. BS 2570.

BSI. 1973b. Natural fibre twines. BS 2570, British Standards Institution, London. 3rd. edn.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. 2020. Efficient intent detection with dual sentence encoders. *arXiv preprint arXiv:2003.04807*.

A. Castor and L. E. Pollux. 1992. The use of user modelling to guide inference and learning. *Applied Intelligence*, 2(1):37–53.

Ashok K. Chandra, Dexter C. Kozen, and Larry J. Stockmeyer. 1981. Alternation. *Journal of the Association for Computing Machinery*, 28(1):114–133.

Ajay Chatterjee and Shubhashis Sengupta. 2020. Intent mining from past conversations for conversational agent. *arXiv preprint arXiv:2005.11014*.

J.L. Chercheur. 1994. *Case-Based Reasoning*, 2nd edition. Morgan Kaufman Publishers, San Mateo, CA.

N. Chomsky. 1973. Conditions on transformations. In *A festschrift for Morris Halle*, New York. Holt, Rinehart & Winston.

James W. Cooley and John W. Tukey. 1965. An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, 19(90):297–301.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Umberto Eco. 1990. *The Limits of Interpretation*. Indian University Press.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231.

Dan Gusfield. 1997. *Algorithms on Strings, Trees and Sequences*. Cambridge University Press, Cambridge, UK.

J Han. 2001. m. kamber (2001): Data mining, concepts and techniques.

Iryna Haponchyk and Alessandro Moschitti. 2021. Supervised neural clustering via latent structured output learning: application to question intents. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3364–3374.

Iryna Haponchyk, Antonio Uva, Seunghak Yu, Olga Uryupina, and Alessandro Moschitti. 2018. Supervised clustering of questions into intents for dialog system applications. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2310–2321.

Michael Heck, Nurul Lubis, Benjamin Ruppik, Renato Vukovic, Shutong Feng, Christian Geishauser, Hsien-Chin Lin, Carel van Niekerk, and Milica Gašić. 2023. Chatgpt for zero-shot dialogue state tracking: A solution or an opportunity?

Paul Gerhard Hoel. 1971a. *Elementary Statistics*, 3rd edition. Wiley series in probability and mathematical statistics. Wiley, New York, Chichester. ISBN 0 471 40300.

Paul Gerhard Hoel. 1971b. *Elementary Statistics*, 3rd edition, Wiley series in probability and mathematical statistics, pages 19–33. Wiley, New York, Chichester. ISBN 0 471 40300.

Yen-Chang Hsu, Zhaoyang Lv, Joel Schlosser, Phillip Odom, and Zsolt Kira. 2019. Multi-class classification without multi-class labels. *arXiv preprint arXiv:1901.00544*.

Otto Jespersen. 1922. *Language: Its Nature, Development, and Origin*. Allen and Unwin.

Hao Lang, Yinhe Zheng, Jian Sun, Fei Huang, Luo Si, and Yongbin Li. 2022. Estimating soft labels for out-of-domain intent detection. *arXiv preprint arXiv:2211.05561*.

Ting-En Lin, Hua Xu, and Hanlei Zhang. 2020. Discovering new intents via constrained deep adaptive clustering with cluster refinement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8360–8367.

Lindung Parningotan Manik, Zaenal Akbar, Hani Febri Mustika, Ariani Indrawati, Dwi Setyo Rini, Agusdin Dharma Fefirenta, and Tutie Djarwaningsih. 2021. Out-of-scope intent detection on a knowledge-based chatbot. *International Journal of Intelligent Engineering & Systems*, 14(5).

OpenAI. 2023. Gpt-4 technical report.

Wenbo Pan, Qiguang Chen, Xiao Xu, Wanxiang Che, and Libo Qin. 2023. A preliminary evaluation of chatgpt for zero-shot dialogue understanding. *arXiv preprint arXiv:2304.04256*.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*.

Hugh Perkins and Yi Yang. 2019. Dialog intent induction with deep multi-view clustering. *arXiv preprint arXiv:1908.11487*.

Jiashu Pu, Guandan Chen, Yongzhu Chang, and Xiaoxi Mao. 2022. Dialog intent induction via density-based deep clustering ensemble. *arXiv preprint arXiv:2201.06731*.

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. 2023. Is chatgpt a general-purpose natural language processing task solver? *arXiv preprint arXiv:2302.06476*.

Mohammad Sadegh Rasooli and Joel R. Tetreault. 2015. Yara parser: A fast and accurate dependency parser. *Computing Research Repository*, arXiv:1503.06733. Version 2.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823.

Chen Shi, Qi Chen, Lei Sha, Sujian Li, Xu Sun, Houfeng Wang, and Lintao Zhang. 2018. Autodialabel: Labeling dialogue data with unsupervised learning. In *Proceedings of the 2018 con-*

ference on empirical methods in natural language processing, pages 684–689.

Charles Joseph Singer, E. J. Holmyard, and A. R. Hall, editors. 1954–58. *A history of technology*. Oxford University Press, London. 5 vol.

Jannik Strötgen and Michael Gertz. 2012. Temporal tagging on different domains: Challenges, strategies, and gold standards. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 3746–3753, Istanbul, Turkey. European Language Resource Association (ELRA).

S. Superman, B. Batman, C. Catwoman, and S. Spiderman. 2000. *Superheroes experiences with books*, 20th edition. The Phantom Editors Associates, Gotham City.

Junyuan Xie, Ross Girshick, and Ali Farhadi. 2016. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pages 478–487. PMLR.

Jiaming Xu, Peng Wang, Guanhua Tian, Bo Xu, Jun Zhao, Fangyuan Wang, and Hongwei Hao. 2015. Short text clustering via convolutional neural networks. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 62–69.

Ka Yee Yeung and Walter L Ruzzo. 2001. Details of the adjusted rand index and clustering algorithms, supplement to the paper an empirical study on principal component analysis for clustering gene expression data. *Bioinformatics*, 17(9):763–774.

Hanlei Zhang, Xiaoteng Li, Hua Xu, Panpan Zhang, Kang Zhao, and Kai Gao. 2021a. Textoir: An integrated and visualized platform for text open intent recognition. *arXiv preprint arXiv:2110.15063*.

Hanlei Zhang, Hua Xu, Ting-En Lin, and Rui Lyu. 2021b. Discovering new intents with deep aligned clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14365–14373.

Yuwei Zhang, Haode Zhang, Li-Ming Zhan, Xiao-Ming Wu, and Albert Lam. 2022. New intent discovery with pre-training and contrastive learning. *arXiv preprint arXiv:2205.12914*.