Graphire: Novel Intent Discovery with Pretraining on Prior Knowledge using Contrastive Learning

Xibin Gao, Radhika Arava, Qian Hu, Thahir Mohamed,
Wei Xiao, Zheng Gao, Mohamed AbdelHady
gxbin,aravar,huqia,thahirm,weixiaow,zhenggao,mbdeamz@amazon.com
Alexa AI
Seattle, Washington, USA

ABSTRACT

In this paper, we introduce Graphire, an intent discovery system leveraging pretraining on predefined intents to automatically discover novel intents for intelligent personal assistants (IPA). In order to transfer the prior knowledge of predefined intents, Graphire first transforms predefined class memberships into pairwise relationships, and then learns a Siamese Neural Network (SNN) model classifying if two utterances have the same intent. The siamese neural network condenses the prior knowledge of predefined intents in the form of trained neural network weights, and infer pairwise relationships among new utterance pairs. The contribution of the paper is three folds: (1) it proposed a pretraining paradigm based on contrastive learning to distill prior knowledge from existing intents. (2) it proposed a new method to discover novel intent leveraging the prior knowledge (3) it proposed a cluster summarization approach to assign labels for the intents. The experimental results demonstrate the effectiveness of pretraining in Graphire and Graphire's capability to discover novel intents on a real-world IPA dataset with intents from disparate domains.

KEYWORDS

intent discovery, contrastive learning, siamese neural networks

ACM Reference Format:
Xibin Gao, Radhika Arava, Qian Hu, Thahir Mohamed,, Wei Xiao, Zheng Gao, Mohamed AbdelHady, . 2021. Graphire: Novel Intent Discovery with Pretraining on Prior Knowledge using Contrastive Learning. In *Proceedings of (Pretrain@KDD).* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Intelligent personal assistants (IPA) such as Amazon Alexa, Apple Siri, Google Assistant, and Microsoft Cortana have gained popularity among users and achieved massive adoption in recent years. These IPAs automatically analyze user utterances and respond accordingly to address user requests. Parsing and understanding the requests from the user utterances is a key task in the natural language understanding components of these IPAs. User requests express *intents* of the users. For example, "play the song despacito" and "play bon jovi songs" express *PlayMusic* intent, while "turn off the hallway light" and "shut the kitchen light" express *TurnOffLight* intent. These IPAs rely on interpretations of the intent carried in the utterances to respond to users in a proper manner.

The research in intent detection from utterances has been prolific [7, 13, 14]. They mostly treat intent detection as a classification task, and rely on large amount of labeled data. Therefore they can only detect predefined intents seen in the training data.

However, users interact with IPAs in various ways and their intents evolve over time. In real world, new intents emerge and old intents may die out. The predefined intents can't cover the ever-changing customer requests and interaction patterns. Therefore large amount of recent utterances become underserved or unserved by IPAs due to their inability to understand the emerged intents. It is crucial to discover new intents from utterances, and hence domain experts can develop new apps to properly address these novel intents. As a result, we can eventually bring the new utterances under the umbrella of service and reduce the customer friction.

A few recent works investigate the discovery of novel intents [31, 32]. Unfortunately, the methods employed ignore the presence of labeled intents. Overall, these existing intent detection and discovery methods bear a few shortcomings and hinder effective capture of novel intent.

- Target distribution is different. The utterances in live traffic may, and are likely to, have different distributions from utterances used in training. Therefore, it renders classification method obsolete as it can't predict intent not present in the training data.
- Prior knowledge is ignored. There is plenty of labeled data with utterance domains and intents information. But unsupervised approach disregard the rich labeled dataset, which may provide clues how new intent can be formed.
- Discovered intent granularity is arbitrary. Due to the subjective nature of intent definitions, there is no gold standard to delineate the granularity. "Read book at normal speed" (carrying *ReadNormal* intent) and "read book faster" (carrying *ReadFaster*) could possibly be merged to *ControlReadSpeed* intent. Existing methods often leave it to a hyperparameter to decide.

To alleviate the challenges above, we propose Graphire, which pretrains with predefined intents and transfer the knowledge to guide the discovery of novel intents from the unlabeled utterances. Graphire first distills knowledge from predefined intents with Siamese Neural Network, a twin neural network with shared parameters and

Pretrain@KDD, August 15th, 2021, Virtual

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/1122445.1122456

a contrastive loss function, trained with pairs of utterances from predefined intents.

Popular in image domain for applications such as face verification [3, 5, 26], localization [17, 29], and object tracking [2, 33], SNN applications in text, as a vehicle to transfer across distributions, have been rare and far between. In our experiments, we show its effectivenes to transfer the knowledge in predefined intents, captured through siamese network model, to the target unlabeled utterance via inferring the pairwise relationships.

After pairwise relationship inference, the unlabeled utterances form an undirected connected graph with nodes representing utterances and edges representing inferred relationships. Finally, Graphire mines for cliques in the graph to discover and summarize the newly formed intents. The goal of pretraining is to transfer the prior knowledge in this paper. Therefore, we use transfer learning and pretraining in a loosely interchangeable manner in this context.

The contributions of this paper are three folds.

- Proposed a transfer paradigm via SNN to transfer knowledge from existing intents to discover new intents.
- Proposed a new framework to discover novel intents from unlabeled utterances in live traffic for IPAs.
- Proposed a graph-based intent extraction approach to assign intent labels to utterance clusters.

Section 2 highlights related work in the research community and illustrates the architectural details of Graphire. Section 3 explains the real-world evaluation dataset, evaluation metrics, and experimental results. Section 4 draws conclusions and points to future work.

2 METHOD

2.1 Related Work

Our work is related to intent detection, contrastive learning, and transfer learning in the research community.

Intent detection. Research in intent detection, particularly with neural network-based approach, is numerous, e.g., Gangadharaiah and Narayanaswamy [7], Goo et al. [8], Kim et al. [13, 14], Kurata et al. [15], Mesnil et al. [18]. These work focus on detecting intents already present in the training data, as opposed to our proposed method to detect novel intents which are not previously defined. In area of novel intent detection, Vedula et al. [31] uses hierarchical clustering to find new intents, while Vedula et al. [32] looks to discover novel intents through the lens of finding *verb* and *object* paired relationships. Both approaches suffer from the shortcomings explained in section 1 in one way or another.

Contrastive learning. Contrastive learning is popular in image domain to tackle a variety of problems such as face verification [3, 5, 26], localization [17, 29], and object tracking [2, 33]. It's also used to learn the object visual representations. For example, Hadsell et al. [10] learn visual features by contrasting positive pairs against negative pairs. Most relevant to our work, Hsu et al. [11] learns a pairwise similarity function for domain and task adaption. They use the transferred knowledge as constraints to guide the clustering in the downstream steps. In text domain, Guo et al. [9] uses contrastive learning to obtain effective representations for text classification based on monolingual embeddings of BERT.

Intent Categorical Membership		**Utterance Pairwise Relationship**		
Utterance	Intent	Utterance	Utterance	Label
u_1	i_1	u_1	u_2	1
u_2	i_1	u_1	u_3	0
u_3	i_2	u_2	u_3	0

Table 1: Transform from categorical intent membership to binary pairwise relationship between utterances.

Transfer Learning. Transfer learning transfers a variety of knowledge such as training instances, features, model parameters and relational knowledge[20]. Pretrained language models like BERT[6] boosted performance of downstream tasks[25, 27], in which the knowledge is transferred through shared features or model parameters. The SNN model with learnt parameters, in this paper, transfers the relational knowledge from labeled data to unlabeled data.

2.2 Problem Definition

Given a set of utterances u_1, u_2, ..., u_n, we aim to assign intents i for each utterance, and eventually it produces (u_1, i_1), (u_2, i_2),...,(u_n, i_k). As prior knowledge, there may exist an auxiliary dataset which has utterances au and corresponding intents ai: (au_1, ai_1), (au_2, ai_2), ..., (au_m, ai_k). Instances of i and ai can be non-overlapping.

2.3 Graphire Overview

To solve the problem above, we propose an approach named Graphire, whose architecture is illustrated in Figure 1 with three steps.

(1) Transfer. First we develop a SNN based transfer paradigm to condense the knowledge from predefined intents. In this stage, the pairwise relationships among the utterances are inferred with the trained siamese network.
(2) Mine. The pairwise relationships among the utterances form an undirected utterance graph, in which vertices represent utterances and edges represent confidences of the connected pairs of utterances having the same intent. We mine for the groups of utterances with the same intent, through the proxy of cliques.
(3) Rank. We generate extractive summaries of groups of utterances as intent labels, through a variant of PageRank[19], intent rank algorithm (IR).

2.4 Transfer

We first transform intent categorical membership to utterance pairwise relationship as shown in Table 1.

With pairwise relationship data, we build a SNN model as shown in Figure 2. Both branches of the network have shared weights with the same f_θ as an encoding scheme. As a result, it produces semantic representations, in the form of fixed-length vectors, for the input pair of utterances. We experiment with a variety of encoding schemes such as GloVe[21], ELMo[22] and BERT[6, 30].

The distances between pairs of encoded vectors are calculated with cosine distance. After that, it applies a contrastive loss function which fine-tunes the underlying semantic space to push embeddings of utterances belonging to the same intent together and pulls embeddings of utterances belonging to different intents apart.

$$L_{\text{contrastive}} = Y * (D_w)^2 + (1 - Y) * (\max(0, m - D_w))^2 \quad (1)$$

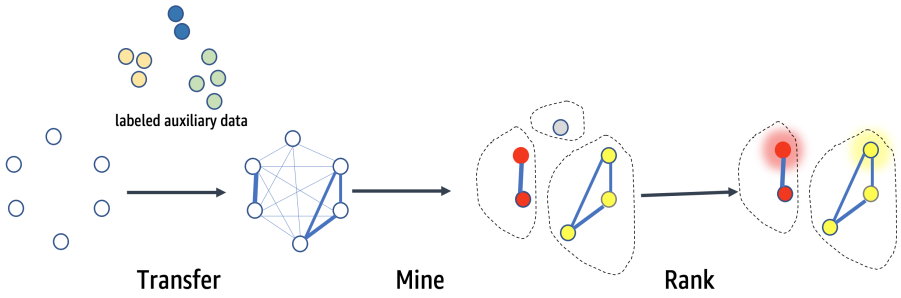

Figure 1: Graphire system. The uncolored circles represents utterances that need to have intents assigned. The labeled auxiliary data means the predefined intents. The solid edges among the circles represent the relationships among the utterances. The enclosing dotted lines delineate intent membership. The glowing nodes indicate the extracted representative utterance in that intent cluster.

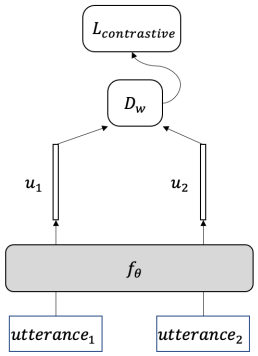

Figure 2: SNN. A Pair of utterances, $utterance1_1$ and $utterance_2$, use the same encoder f_θ with shared weights. The encoder generates a pair of vectors u_1 and u_2 for the input pair of utterances independently. A distance (or similarity) function D_w is applied on the pair of vectors. The network is optimized with contrastive loss function $L_{\text{contrastive}}$

In contrastive loss function shown in equation 1, the margin term m is used to tighten the constraint. If two utterances in a pair are not the same intent, then their distance should be at least margin, or a loss will be incurred. $Y = 0$ if utterances are from different intents; else $Y = 1$ if the utterances are from the same intent. Learnable distance function $D_w(x_1, x_2)$ between a pair of utterances x_1, x_2 is parameterized by the weights W in the neural network. Specifically in our implementation, $D_w(x_1, x_2)$ is a network transformation on the cosine distance. The network concurrently optimize for D_w and the encoder parameters in f_θ.

SNN is trained the labeled auxiliary data to obtain the pairwise relationship prediction model, which is later used to infer pairwise relationships among the unlabeled utterances.

2.5 Mine

The inferred pairwise relationships form a graph with vertices representing utterances and edges representing the likelihood the connected vertices belong to the same intent. If the weight between two edges is greater than the default threshold 0.5, it is regarded as connnected, otherwise disconnected. We mine the undirected weighted graph for cliques. A clique, C, in an undirected graph $G = (V, E)$ is a subset of the vertices, $C \subseteq V$, such that every two distinct vertices are adjacent[1]. In utterance graph, the edges represent the confidences of the connected vertices being the same intent. Therefore, a clique forms a group of coalesced utterances with the same intent.

This implementation of clique finding algorithm we applied is based on work by Bron and Kerbosch [4] and unrolls the recursion to avoid issues of recursion stack depth, as in adaption by Tomita et al. [28]. With the utterance cliques uncovered, we regard cliques with size smaller than k as noise, and hence discard them.

2.6 Rank

We developed a PageRank[19] inspired algorithm, IntentRank, to label each clique with its contained utterance. In weighted undirected graph to obtain intent rank (IR) for each utterance u within a clique with equation 2.

$$\text{IR}(u) = \sum_{v \in B_u} \text{IR}(v) * w(u, v) \tag{2}$$

where B_u are neighbors of utterance u and $w(u, v)$ is the edge weight connecting two vertices u and v in the graph, or the likelihood of utterance u and utterance v having the same intent. Eventually, the highest ranked utterance per clique is returned as the extractive summary of the intent cluster.

3 EXPERIMENTS

3.1 Dataset

We apply a real-world dataset from IPA users to evaluate the performance of Graphire. In actual scenarios, labeled data may only have a few utterances per intent due to the lack of resources or the ad-hoc nature of annotation. In our evaluation, we are also using intents with small number (ranging from dozens to a few hundred) of utterances from a disparate domains such as education, recipe, shopping, video, book, calendar, health, and music.

In the training phase, the intent labels are used to generate utterance pairwise labels. In the testing phase, the intents are hidden

Task	Dataset	Ratio
In-Domain	Training	16:1
	Training after sampling	2:1
	Testing	14:1
Cross-Domain	Training	15:1
	Training after sampling	2:1
	Testing	13:1

Table 2: Pairwise Relationship Dataset. For user data privacy and intellectual property reasons, we don't show the exact number of pairs. Ratio indicates the proportion of the number of negative pairs to the number of positive pairs.

and the model predictions are compared with the hidden intents to compute model quality metrics.

To better evaluate the efficacy of the approach, we formulate two evaluation tasks: in-domain intent discovery and cross-domain intent discovery.

- In-domain intent discovery: training data and testing data are from the same domains. Their intents are mutually exclusive. The dataset is produced with random sampling and de-identification to protect user privacy. Then it is split to train (8 domains, 26 intents) and test (8 domains, 21 intents).
- Cross-domain intent discovery: training data and testing data are from different domains. Their intents are mutually exclusive. The dataset is produced with random sampling and de-identification to protect user privacy. Then it is split to train (4 domains, 28 intents) and test (4 domains, 19 intents).

We divide the evaluation into two stages: the pairwise relationship prediction evaluation, and intent discovery evaluation.

3.2 Pairwise Relationship Evaluation

In this section, we evaluate the performance of the pairwise relationship prediction. We measure the performance of the SNN and its ability to transfer knowledge from labeled intents to the test set. The evaluation metrics are below.

- Precision: The fraction of relevant instances among the retrieved instances.
- Recall: The fraction of retrieved relevant instances among all relevant instances.
- F-measure: The harmonic mean between precision and recall.

Data sampling. The training data of the pairwise relationship prediction task is highly imbalanced as the number of negative instances is over ten times higher than the number of positive instances. To improve the overall model performance, we downsampled the negative instances to the same scale as the number of positive instances in the training data while leaving testing data unchanged as shown in Table 2.

Baseline method. Instead of using transfer learning via SNN, the baseline method applies content-based cosine similarity measures to obtain the pairwise relationships. From the training set, it computes the cosine similarity between pairs of BERT-encoded utterances and obtains threshold where it achieves maximal F-1 measure, and then it applies this threshold to decide, in the test set, if pairs of utterances are positive or negative based on their cosine similarities.

Encoders. We experiment with a variety of encoders f_θ in the SNN including pretrained GloVe[21], ELMo[22] and BERT[6, 30]. For a sequence of tokens, we apply either recurrent or convolutional stacks on top of the GloVe and ELMo embeddings to obtain utterance level encoding. The convolutional stack uses three filters of size 256 and max pooling as shown in Figure 3. The recurrent stack uses bidirectional LSTM with hidden dimension size of 200 and mean pooling as shown in Figure 4.

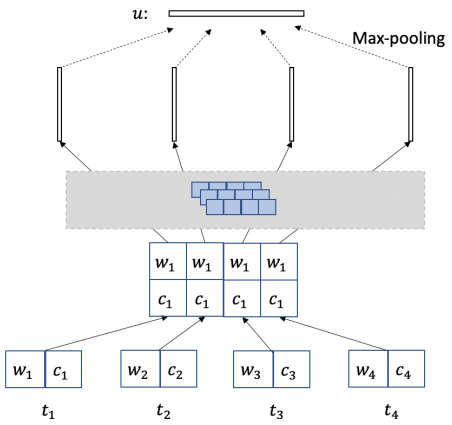

Figure 3: Convolutional stack. For each token t_i in utterance, it derives its word embedding w_i and character embedding c_i and then concatenate them together. Convolutional layers as shown in the gray box are applied on top of the concatenated vector and eventually max pooling is applied to get sentence vector u.

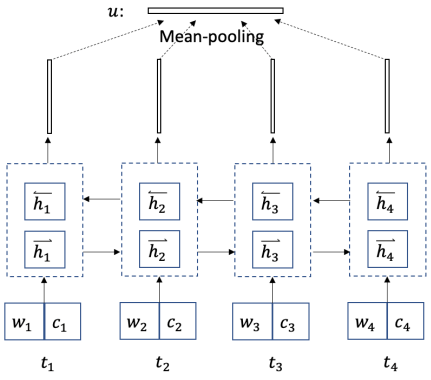

Figure 4: Recurrent stack. For each token t_i in utterance, it derives its word embedding w_i and character embedding c_i and then concatenate them together. BiLSTM is applied on top of the concatenated vector and eventually mean pooling is applied to get sentence vector u.

The results are shown in Table 3. For both cross-domain and in-domain tasks, it shows that transfer learning with BERT encoder outperforms baseline method, which calculates the cosine similarity over the utterance BERT embeddings. Furthermore, we observe that pretrained GloVe and ELMo embeddings, outperforms fine tuned BERT. It is consistent with the results from Li et al. [16], Reimers and Gurevych [23], where it is shown that BERT encoder does not capture semantics well. Additionally, ELMo has advantage over GloVe all experimental setups. On top of the ELMo and GloVe, convolutional or recurrent layers(s) generate utterance level embedding,

Task	Model	Prec.	Rec.	F-1	AUC
In-Domain	BERT cosine baseline	-	-	-	-
	Siamese+BERT	+0.31	+0.08	+0.32	+0.28
	Siamese+GloVe+CNN	+0.12	+0.22	+0.17	+0.26
	Siamese+GloVe+RNN	+0.26	+0.26	+0.32	+0.33
	Siamese+ELMo+CNN	**+0.42**	+0.23	**+0.45**	+0.35
	Siamese+ELMo+RNN	+0.35	**+0.26**	+0.41	**+0.36**
Cross-Domain	BERT cosine baseline	-	-	-	-
	Siamese+BERT	+0.08	-0.01	+0.10	+0.14
	Siamese+GloVe+CNN	+0.04	**+0.21**	+0.07	+0.18
	Siamese+GloVe+RNN	+0.17	+0.17	+0.22	+0.26
	Siamese+ELMo+CNN	**+0.25**	+0.07	+0.27	+0.25
	Siamese+ELMo+RNN	+0.30	+0.20	**+0.35**	**+0.32**

Table 3: Pairwise relationship prediction performance. For intellectual property reasons, we report the absolute gains (or losses) of the models over the baseline method. BERT cosine baseline refers to the model uses cosine similarity between BERT encodings. Siamese+BERT method keeps byte pair encoding frozen while tuning positional encoding and the rest of the network with 50 epochs. GloVe and ELMo based models are fined tuned on the CNN (or RNN) stack with 50 epochs.

Task	Model	S	Hom.	Com.	V-m.	Int. Rec.
In-Domain	K-means baseline	-	-	-	-	-
	Graphire+Best	1	**+0.35**	+0.04	+0.15	-0.05
	Graphire+Best	5	+0.28	+0.16	+0.22	-0.24
	Graphire+Best	10	+0.28	+0.24	+0.26	-0.33
	Graphire+Best	15	+0.30	**+0.31**	**+0.30**	-0.38
Cross-Domain	K-means baseline	-	-	-	-	-
	Graphire+Best	1	**+0.42**	-0.02	+0.13	+0.01
	Graphire+Best	5	+0.26	+0.03	+0.12	**+0.11**
	Graphire+Best	10	+0.12	+0.11	+0.12	-0.51
	Graphire+Best	15	+0.36	**+0.23**	**+0.29**	-0.32

Table 4: Intent discovery performance. For intellectual property reasons, we report the absolute gains (or losses) of the models over the baseline method. S means minimum cluster size. Hom means homogeneity. Com. means completeness. V-m. means V-measure. Int. Rec. means intent recall. Graphire+Best means Graphire configured with the best performing SNN model: siamese+ELMo+RNN in cross-domain task and siamese+ELMo+CNN in in-domain task.

and it shows that with the same embedding scheme, recurrent stack, in most cases, outperforms convolutional stack.

3.3 Intent Discovery Evaluation

In this section, we evaluate the performance of end-to-end intent discovery of Graphire, with a set of clustering-based metrics from [24] and custom defined metrics.

- Homogeneity: Entropy based measure. A clustering result satisfies homogeneity if all of its clusters contain only data points which are members of a single class.
- Completeness: Entropy based measure. A clustering result satisfies completeness if all the data points that are members of a given class are elements of the same cluster.
- V-measure: A harmonic mean of homogeneity and completeness.
- Intent recall: The fraction intents that are discovered. In the evaluation set, each predicted cluster is labeled with its dominant intent. Collecting over all the dominant intents from the predicted clusters will give the total number of discovered intents.

Homogeneity, completeness, and V-measure are general metrics for clustering algorithms, while intent recall is a metric specific to intent discovery, which is used as an auxiliary measure and should be viewed along side with clustering-based metrics.

Baseline method. With unsupervised learning, we use FAISS implementation [12] of k-means to generate clusters of utterances as baseline. The clustered utterances are regarded as having the same intent. In reality, the number of intents is unknown. To get the best of baseline approach results, we set k as the number of intents in the test set.

Denoise with clique size. Very small cliques rarely form useful intents. Therefore, we experiment with different cutoff thresholds S for the minimum clique size. As the minimum clique size grows, the performance is expected to improve.

The results are shown in Table 4. We observe that in both cross-domain and in-domain tasks, Graphire beats baseline approach in homogeneity, completeness, and V-measure. Graphire slightly underperforms in intent recall in in-domain task. One possible explanation is that the efficacy of knowledge transfer is more evident when the task is more difficult, as cross-domain task is considered more challenging than in-domain task.

It is also observed that as a general trend, when the minimum clique size increases, homogeneity decreases and completeness increases. It is reasonable that in the denoising process, Graphire gradually discards cliques of small sizes. As the cutoff threshold in the output increases, the number of cliques left tend to decrease, resulting in higher completeness score. On the other hand, larger cliques tend to contain utterances of different intents, therefore it results in lower homogeneity score. There are two most extreme cases. 1) Each utterance forms its a separate clique. Then the homogeneity score is of a high value while the completeness score is of a low value. 2) All the utterances form a single clique. Then the homogeneity score is of a low value while the completeness score is of a high value. As a comprehensive measure and a tradeoff between homogeneity and completeness, V-measure increases as the minimum clique size threshold increases, showing the promise of denoising capability of mimum clique size parameter.

It's worth mentioning that the k-means baseline method need to tune the hyper parameter k and it's often challenging to estimate the number of intents in live traffic. However, graph-based Graphire doesn't need to know in prior how many intents there are in the data to be inferred. Therefore it render Graphire applicable in many real world scenarios.

3.4 Challenges

Challenges abound in the intent discovery process. Transforming categorical intent membership to binary pairwise relationship can grow the instances quadratically. Intent membership for n utterances can map up to n^2 pairwise relationships. One mitigation technique is to sample the utterances from the whole population and then generate their pairs, since we are targeting to find the novel intent and not necessarily include all the utterance instances of the intents. Another way to reduce the data load (in training) is to explore in-batch pair generation, producing positive and negative pairs with utterances only from within batch. The clique finding algorithm we applied[28] may find overlapping cliques, e.g., a utterancen u belong to both clique C_1 and C_2. Graphire implementation assigns u to either C_1 or C_2 with a random seed and make the cliques mutually exclusive.

4 CONCLUSION

In this paper, we proposed Graphire, a method to automatically discover novel intents from utterances building on top of pretaining on existing intents. First, we transformed the categorical knowledge

from predefined intents to binary pairwise relationship between labeled utterances and condenses the knowledge to a siamese neural network model. With contrastive loss function, the network fine tunes the featurizer (f_θ) and projects the encoding vectors of utterances with the same intent closer to each other while pushing those with different intents further apart. With the fine tuned siamese neural network, it predicts the pairwise relationships among utterances where new intent abounds. With the undirected weighted graph at hand, we mined for strongly connected components—cliques as coalesced utterances groups that form intents. Further more, we developed intent rank, an adaption of PageRank to discover the most representative utterance as the extractive summary and intent label for each clique. The experiments show, building on pretraining with existing intents, the promise and viability of Graphire in discovering new intents on a real-world dataset.

5 ACKNOWLEDGMENTS

We thank Ryan Gabbard, Beiye Liu, and Melanie Rubino for their valuable opinions on this paper.

REFERENCES

[1] Richard D Alba. 1973. A graph-theoretic definition of a sociometric clique. *Journal of Mathematical Sociology* 3, 1 (1973), 113–126.

[2] Luca Bertinetto, Jack Valmadre, João F. Henriques, Andrea Vedaldi, and Philip H. S. Torr. 2016. Fully-Convolutional Siamese Networks for Object Tracking. In *Computer Vision - ECCV 2016 Workshops - Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part II (Lecture Notes in Computer Science)*, Gang Hua and Hervé Jégou (Eds.), Vol. 9914. 850–865. https://doi.org/10.1007/978-3-319-48881-3_56

[3] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. 1993. Signature Verification Using a Siamese Time Delay Neural Network. In *Advances in Neural Information Processing Systems 6, [7th NIPS Conference, Denver, Colorado, USA, 1993]*, Jack D. Cowan, Gerald Tesauro, and Joshua Alspector (Eds.). Morgan Kaufmann, 737–744. http://papers.nips.cc/paper/769-signature-verification-using-a-siamese-time-delay-neural-network

[4] Coen Bron and Joep Kerbosch. 1973. Algorithm 457: finding all cliques of an undirected graph. *Commun. ACM* 16, 9 (1973), 575–577.

[5] Sumit Chopra, Raia Hadsell, and Yann LeCun. 2005. Learning a Similarity Metric Discriminatively, with Application to Face Verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA*. IEEE Computer Society, 539–546. https://doi.org/10.1109/CVPR.2005.202

[6] 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* (2018).

[7] Rashmi Gangadharaiah and Balakrishnan Narayanaswamy. 2019. Joint multiple intent detection and slot labeling for goal-oriented dialog. 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)*. 564–569.

[8] Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. 753–757.

[9] Zhiqiang Guo, Zhaoci Liu, Zhenhua Ling, Shijin Wang, Lingjing Jin, and Yunxia Li. 2020. Text Classification by Contrastive Learning and Cross-lingual Data Augmentation for Alzheimer's Disease Detection. In *Proceedings of the 28th International Conference on Computational Linguistics*. 6161–6171.

[10] Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, Vol. 2. IEEE, 1735–1742.

[11] Yen-Chang Hsu, Zhaoyang Lv, and Zsolt Kira. 2018. Learning to cluster in order to transfer across domains and tasks. In *International Conference on Learning Representations*.

[12] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. Billion-scale similarity search with GPUs. *arXiv preprint arXiv:1702.08734* (2017).

[13] Joo-Kyung Kim, Gökhan Tür, Asli Çelikyilmaz, Bin Cao, and Ye-Yi Wang. 2016. Intent detection using semantically enriched word embeddings. In *2016 IEEE Spoken Language Technology Workshop, SLT 2016, San Diego, CA, USA, December 13-16, 2016*. IEEE, 414–419. https://doi.org/10.1109/SLT.2016.7846297

[14] Young-Bum Kim, Sungjin Lee, and Karl Stratos. 2018. OneNet: Joint Domain, Intent, Slot Prediction for Spoken Language Understanding. *CoRR* abs/1801.05149 (2018). arXiv:1801.05149 http://arxiv.org/abs/1801.05149

[15] Gakuto Kurata, Bing Xiang, Bowen Zhou, and Mo Yu. 2016. Leveraging Sentence-level Information with Encoder LSTM for Semantic Slot Filling. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2077–2083.

[16] Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. *arXiv preprint arXiv:2011.05864* (2020).

[17] Tsung-Yi Lin, Yin Cui, Serge J. Belongie, and James Hays. 2015. Learning deep representations for ground-to-aerial geolocalization. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. IEEE Computer Society, 5007–5015. https://doi.org/10.1109/CVPR.2015.7299135

[18] Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, et al. 2014. Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23, 3 (2014), 530–539.

[19] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. *The PageRank citation ranking: Bringing order to the web.* Technical Report. Stanford InfoLab.

[20] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22, 10 (2009), 1345–1359.

[21] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*. 1532–1543. http://www.aclweb.org/anthology/D14-1162

[22] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL-HLT*. 2227–2237.

[23] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 3973–3983.

[24] Andrew Rosenberg and Julia Hirschberg. 2007. V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Association for Computational Linguistics, Prague, Czech Republic, 410–420. https://www.aclweb.org/anthology/D07-1043

[25] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to Fine-Tune BERT for Text Classification? *arXiv* (2019), arXiv–1905.

[26] Yi Sun, Xiaogang Wang, and Xiaoou Tang. 2014. Deep Learning Face Representation from Predicting 10, 000 Classes. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*. IEEE Computer Society, 1891–1898. https://doi.org/10.1109/CVPR.2014.244

[27] Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT Rediscovers the Classical NLP Pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 4593–4601.

[28] Etsuji Tomita, Akira Tanaka, and Haruhisa Takahashi. 2006. The worst-case time complexity for generating all maximal cliques and computational experiments. *Theoretical computer science* 363, 1 (2006), 28–42.

[29] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christoph Bregler. 2015. Efficient object localization using Convolutional Networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. IEEE Computer Society, 648–656. https://doi.org/10.1109/CVPR.2015.7298664

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[31] Nikhita Vedula, Rahul Gupta, Aman Alok, and Mukund Sridhar. 2020. Automatic Discovery of Novel Intents & Domains from Text Utterances. *arXiv preprint arXiv:2006.01208* (2020).

[32] Nikhita Vedula, Nedim Lipka, Pranav Maneriker, and Srinivasan Parthasarathy. 2020. Open Intent Extraction from Natural Language Interactions. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, Yennun Huang, Irwin King, Tie-Yan Liu, and Maarten van Steen (Eds.). ACM / IW3C2, 2009–2020. https://doi.org/10.1145/3366423.3380268

[33] Zheng Zhu, Qiang Wang, Bo Li, Wei Wu, Junjie Yan, and Weiming Hu. 2018. Distractor-Aware Siamese Networks for Visual Object Tracking. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part IX (Lecture Notes in Computer Science)*, Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss (Eds.), Vol. 11213. Springer, 103–119. https://doi.org/10.1007/978-3-030-01240-3_7