Information Sciences 689 (2025) 121493

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

A linguistics-based approach to refining automatic intent detection in conversational agent design

Alessandra Ferrera [a,c], Giulio Mezzotero [b,c], Domenico Ursino [b,*]

[a] FICLIT, University of Bologna, Italy
[b] DII, Polytechnic University of Marche, Italy
[c] Dinova Srl, Italy

ARTICLE INFO

Keywords:
Automatic intent detection
Conversational agents
Embedding
Dimensionality reduction
Clustering
Labeling

ABSTRACT

In this paper, we propose Automatic Intent Detector (AID), a framework for automatic intent detection to facilitate the creation of a conversational agent. AID follows an eight-step process incorporating best practices from the current literature and introducing innovative approaches in certain steps. The most notable innovation within AID is the automatic labeling of clusters, which is based on detailed and sophisticated rules derived from linguistics. These rules focus on morphosyntactic analysis, while also taking into account an aspect of semantic role theory. Furthermore, as for the overall validation of the results obtained, it provides an approach based on the concepts of semantic coherence, variability, and label appropriateness. After describing AID at the technical level, we illustrate the experiments we conducted both on a dataset widely used as benchmark in the literature and on a real corporate dataset. Finally, we present a critical discussion on the results obtained.

1. Introduction

In recent years, we have witnessed a great proliferation of conversational agents and studies about human-bot interaction [42,17] in a variety of contexts where they have assumed increasingly significant roles. Some possible applications of such agents, which are still largely being explored, include health care [3,40], education [21,9], Public Administration [28], and corporate sectors. In particular, in the latter area, conversational agents are being used for marketing and e-commerce [22], external customer support [45], and internal business process optimization [37]. This growing popularity is due to the many benefits that conversational agents bring, such as the ability to simultaneously serve a greater number of users and to offer immediate responses without service interruption.

Most conversational agents currently available are based on the Intent-Entity-Context-Response (IECR) paradigm. This paradigm combines a Natural Language Understanding (NLU) component, which aims to identify intents and extract entities, with predefined answers from a conversational designer. Despite the emergence of new technologies that promise to revolutionize this field in the incoming years, IECR paradigm has advantages that are still fundamental. Indeed, it guarantees an absolute control over every aspect of answers. First and foremost, this allows for total accuracy from a content perspective, which is particularly crucial in those contexts where the information provided by a conversational assistant consists of legal matters (think, for instance, of chatbots for Public Administration) or concerns the health of users. Second, the IECR paradigm allows control over the stylistic and linguistic

* Corresponding author.
E-mail addresses: alessandra.ferrera@dinova.one (A. Ferrera), giulio.mezzotero@dinova.one (G. Mezzotero), d.ursino@univpm.it (D. Ursino).

https://doi.org/10.1016/j.ins.2024.121493
Received 4 March 2024; Received in revised form 12 September 2024; Accepted 15 September 2024

Available online 20 September 2024

levels, adapting the tone-of-voice to the target audience and client identity, while preventing the biases to which generative AI is often exposed [14].

This paper falls right into that context and aims to define Automatic Intent Detector (AID), a framework for simplifying the design of an IECR-based conversational agent, particularly in the construction of intents. As the name suggests, the uniqueness of our approach lies in its ability to perform the intent detection automatically. There are many examples of supervised intent detection in the literature. Such approaches, once the model is trained, are certainly preferable for categorizing incoming requests using a list of pre-existing intents. However, when such pre-existing lists are not available, the goal shifts to identifying new intents from scratch: in this scenario, an automatic intent detection approach seems to be the natural choice. This is especially true in the early stages of conversational agent design, when a large amount of often unlabeled sample text data from very heterogeneous sources (e.g., requests sent to service personnel, mailbox history, web pages with FAQs posed to the customer service department) needs to be analyzed. In this case, even a relatively small number of samples requires a significant effort to manually identify intents, since a human expert would have to analyze each sample individually. To address this problem, AID implements an approach that automatically identifies the intents underlying available data by grouping similar requests and assigning consistent labels to them. Using AID, the workflow of the design of a conversational agent consists of two steps, namely: *(i)* automatic and unsupervised analysis of the source data to optimize the most time-consuming steps in the process; *(ii)* refinement of the obtained intents and formulation of the answers, to be performed manually to ensure a high accuracy of the final result.

The approach underlying AID begins by constructing numerical representations of source texts. For this purpose, it uses Universal Sentence Encoder [6] and Sentence-BERT models [36]. Then, it solves an optimization problem that, by minimizing a score function, detects an ideal combination of parameters in order to perform intent clustering. Once categories are formed, AID defines syntactic rules based on Universal Dependencies formalism [35] to generate automatic labels that best represent the identified intents. This process, which uses the spaCy parser, it the most innovative aspect of our approach. It is precisely the automatic labeling that is the most innovative part of our approach. In fact, it starts from the observation that utterances tend to feature an “action-object” type structure [26]. Here, the “action” component identifies a command, a goal to be pursued, or a task to be performed, while the “object” component represents the entity on which the “action” produces an effect. AID extracts the most frequent “action-object” pairs from each cluster and automatically derives labels from them. To achieve this, it performs a very thorough linguistic analysis, using the PoS tagger, to identify parts of speech, and the dependency parser, trained on the Universal Dependencies corpus, to define syntactic relationships. The analysis performed by AID is morphosyntactic, while maintaining a consideration of the semantic role theory.

We initially tested AID on Banking77 [5], a public dataset ideal for our purposes. In fact, it has fine-grained, well-formulated and equally distributed intents and a set of labels (ground truth) useful for result evaluation. For the validation of AID on such dataset, we employed extrinsic and intrinsic metrics, along with a manual evaluation. Next, we tested AID on a corporate dataset, which required addressing some additional challenges related to: *(i)* the absence of a ground truth, *(ii)* higher query complexity, and *(iii)* inhomogeneity of linguistic data. To address these difficulties, we had to perform preprocessing on the dataset in order to achieve satisfactory categorization. The results obtained from Banking77 were an important reference point when evaluating those from the corporate dataset. Instead, the latter showed the extent to which AID is capable of adapting to the complexity elements of a real dataset.

In summary, the main contributions of this paper are as follows:

- It presents AID, a framework for automatically detecting the intents to which a conversational agent should be able to answer.
- It proposes a method for automatically labeling clusters of sentences in a dataset. This method returns labels that can best capture the content of the cluster and best explicate the underlying intent.
- It proposes some metrics for validating AID and uses them to perform tests on both an ideal dataset and a real one.

The outline of this paper is as follows: Section 2 presents related literature. Section 3 describes the approach underlying AID. Section 4 illustrates the experiments we carried out to evaluate the performance of AID. Finally, Section 5 presents some conclusions that can be drawn about AID and outlines some possible future developments.

2. Related literature

Improving the understanding of user intents in human-computer interaction is a very active research strand, with a wide range of advanced solutions in the past and current literature. Intent detection approaches for conversational agents fall into two categories, namely supervised and unsupervised approaches. Although AID falls into the second category, in this section we also briefly examine supervised approaches, as understanding how they exploit ground truth can be useful in defining the way of proceeding of unsupervised approaches, where, by definition, such ground truth does not exist.

In the field of supervised approaches, in [8] the authors propose an intent detection approach based on Naive Bayes and SVM classifiers, obtaining good F1-score values despite the noisy nature of the data employed for its test. Next, they use the principle of transfer learning as part of their approach. This principle, which aims to recognize similarly expressed intents in different domains, correlates with our approach, aimed at recognizing similar intents expressed in different forms. In [49], the authors propose the IntentBERT model for identifying few-shot intents, and evaluate the performance of logistic regression classifiers. Their pre-training methodology, based on three different public datasets (Banking77, MCID e HINT3), has a higher accuracy than that of the baseline models (CONVBERT, TOD-BERT, USE-ConveRT, DNNC, WikiHowRoBERTa). Other supervised approaches use various kinds of neural network, including convolutional [18,46] and recurrent [25] networks. In general, the adoption of deep learning methods tends to

produce better results in terms of Precision, Recall, and F1-score than those returned by traditional machine learning methods used as baseline.

Regarding unsupervised intent detection approaches, there is a variety of vectorization and clustering methods in the literature, ranging from partitioning, hierachical, and density-based algorithms to, more occasionally, fuzzy logic. In [41], the authors propose a framework to automatically cluster intents and slots in ATIS, a corpus of spontaneous speech in a dialogic form. To achieve their goal, they collect a set of context features, exploit an autoencoder to assemble these features, and employ a dynamic hierarchical clustering method. Their tests show a good degree of clustering accuracy (84.1%), although the peculiarities of the dataset (which contains disfluencies and colloquialisms typical of oral speech) partially compromise the cleanliness of available data.

In [5], the authors present intent detection methods supported by pretrained dual sentence encoders, such as `USE` (Universal Sentence Encoder, previously presented in [6]), and ConveRT, demonstrating their usefulness and wide applicability to three different datasets. Both models have several advantages in that: *(i)* they outperform intent detectors based on fine-tuning the entire BERT-Large model or using BERT as a fixed black-box encoder; *(ii)* they can be quickly trained on a single CPU; and *(iii)* they are stable across different hyperparameter configurations. In addition, the authors of [5] release the Banking77 dataset, which has been used in many studies and approaches on this topic, and which we also adopted in our tests. The authors of [48] propose UNISD, a centroid-driven clustering framework for intent detection. UNISD is based on approaches that allow the inclusion of partially labeled (semi-supervised) data in addition to unlabeled data. It has been tested on Banking77, and an interesting insight that emerged from these tests is the evaluation of the degree to which the considered metrics vary against the percentage of the classification of the dataset used for the training step.

In [12], the authors propose a methodology for comparing embeddings from labeled datasets as part of user feedback analysis. This approach is applied to evaluate both embedding techniques reported in the literature and the most advanced deep text embedding models currently available. The results of [12] confirm the ones of [6] about the effectiveness of `USE` in clustering feedbacks with similar requirements and relevant features. For this reason, `USE` is one of the embedding models we evaluated in our study.

In [43], the authors analyze the performance of BERT in combination with four clustering algorithms, namely K-Means, Eigenspace-based Fuzzy C-Means, Deep Embedded, and Improved Deep Embedded. The results show that BERT outperforms the TF-IDF method in 28 of the 36 metrics used for validation, proving to be an excellent model for text embeddings. An important difference between the approach of [43] and ours is that the former focus on document-based datasets (AG News, Yahoo! Answers, Reuters) whereas ours operates on user-agent dialogues. In [7], the authors propose an intent detection approach based on three key features, namely: *(i)* pre-training of an own language model; *(ii)* use of a multi-task cosine softmax loss function; and *(iii)* embedding and clustering. The experiments of the authors show significant improvements in clustering accuracy over standard sentence embedding approaches. However, due to domain-adapted pre-training, the resulting sentence encoder is no longer generic, and thus may not be applicable to data outside the use cases on which it was trained; moreover, the training process is supervised. For this reason, our approach differs considerably from the one of [7]. However, we borrow from it the idea of using ARI (Adjusted Rand Index) and NMI (Normalized Mutual Information) as metrics for experimental evaluation.

In [30], the authors propose ELDA (Embedding-based Latent Dirichlet Allocation), a new technique based on the combination of topic modeling LDA and sentence embedding. ELDA goes beyond traditional methods by extracting key utterances, and no longer just key words, from corpora of customer-agent dialogues. Working at the sentence level, ELDA solves some of the major bag-of-word problems of traditional LDA. In fact, the recognition of the entire utterance allows the interpretation of words in their context, producing differentiated analyses from homomymous and polysemous words. As a consequence, the topic descriptions produced are more meaningful and easily interpretable, reducing the need to manually review dialogue transcripts. Despite these significant improvements over classical LDA, ELDA retains one feature of it that is not compatible with our use case, namely the need to specify in advance the number of topics to be tracked in the dataset. Also in [2], the author starts from an analysis of traditional topic modeling methods (LDA and Probabilistic Latent Semantic Analysis) to define an approach that bridges their weaknesses. The proposed approach involves vectorizing texts with the DBOW model [24] of top2vec. The vectors generated by this model, originally with 300 dimensions, are reduced with the UMAP (Uniform Manifold Approximation and Projection) algorithm [32] and clustered with HDBSCAN [31]. Some of the methodological insights proposed in [2] are particularly useful and can be taken up by our approach. In particular, dimensionality reduction with UMAP is useful in contrasting the “curse of dimensionality” problem [1], while the use of HDBSCAN is suitable to our scenario, as it is density-based and does not require a predetermined number of clusters to be identified. In [16], the authors propose a topic model that extends the document clustering task by extracting a consistent representation of topics through the development of a class-based variant of TF-IDF. This model also uses UMAP in combination with HDBSCAN, but in this case the embedding step is performed through Sentence BERT [36], which is very effective in converting sentences and paragraphs into dense vector representations. Although this approach is used for document clustering, it can still be used in our context, which is more related to sentence clustering.

In [29], the authors propose an approach that first creates labeled clusters by applying K-Means, agglomerative clustering and LDA. Then, it applies transfer learning and zero-shot learning techniques to detect intents. The analysis of the results shows that agglomerative clustering generates better clusters than K-Means and LDA (partly confirming the effectiveness of hierarchical clustering already demonstrated in [41]). The pre-processing approach proposed in [29] is more invasive than other approaches because it involves the removal of stop words and “unwanted” words, followed by the vectorization of trigrams and quadrigrams exceeding a certain frequency threshold.

In [10], the authors propose Zero-Shot-BERT-Adapters, a method for multilingual intent discovery based on a fine-tuned transformer architecture with adapters. They train the Natural Language Inference (NLI) model and then perform multilingual unknown intent classification in a zero-shot setting. To achieve this, they first analyze the quality of the model after adaptive fine-tuning on

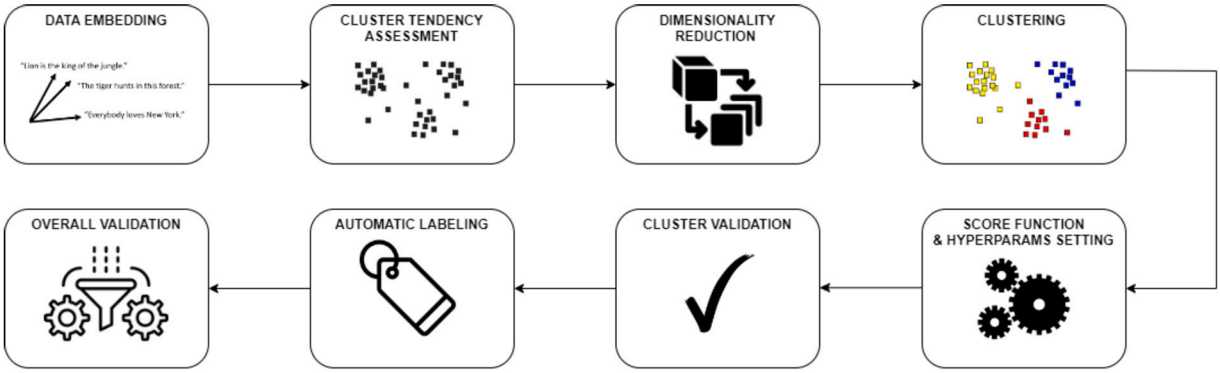

Fig. 1. Steps comprising AID.

known classes. Then, they evaluate its performance in casting intent classification as an NLI task. Finally, they test its zero-shot performance of unknown classes, showing that it can indeed perform intent discovery by generating intents that are semantically similar, if not identical, to the ground-truth ones. This approach confirms several insights, such as the adoption of the USE model for embedding, the evaluation of results using the Banking77 dataset, and the model's ability to work with both labeled and unlabeled data, analogous to what was seen in [48]. The limitations of a cross-lingual approach are also highlighted in this paper.

In [34], the authors evaluate the effectiveness of different textual representations for annotating unlabeled dialogue data. Their methodology includes classical approaches and the adoption of pre-trained transformers for word embeddings. The obtained word embedding results are then employed to create sentence embeddings using pooling methods. Similar to [2], the authors of [34] propose dimensionality reduction by means of UMAP. Despite the use of a more compact data representation, this algorithm improves the performance on embeddings obtained by BERT by up to 20%, outperforming t-SNE [44]. Finally, the sentence embeddings resulting from pooling are fed into clustering algorithms (agglomerative K-Means and HDBSCAN) for intent identification. The results show that transformer-based text representation is superior to that obtained through classical approaches.

To conclude this roundup of related approaches, we mention an automatic labeling method proposed in [26]. It extracts the action-object pair from each utterance using a dependency-based parser. Then, it selects the most frequent pair in each cluster as the representative one for the cluster. The authors evaluated their approach on the SNIPS dataset, obtaining excellent results. In particular, all intent labels were correctly assigned with respect to the ground truth, and the overall F1-score was equal to 93%. Finally, the authors of [19] introduce a two-stage neural embedding framework with redundancy-aware graph-based topic label ranking. The approach is based on a continuous neural vector representation that captures syntactic and semantic information about topic terms and sentences. It then performs an optimization using the topic ranking methods based on the top 20 captured terms. This reduces the computational burden and selects the best candidate sentences with stronger correlation and consistency, further improving the quality of subsequent topic label generation. Based on relevance, coverage, and discrimination attributes, the experimental results show that this approach can extract salient sentences and generate meaningful topic labels with minimal redundancy. These methods provide an interesting starting point for our labeling system.

3. Description of the proposed framework

In this section, we provide a technical description of AID. Our framework takes into account the extensive variety of real-world use cases: in fact, source datasets can differ significantly in terms of size, distribution, content, language, and data noise. Many of the proposed approaches used public and widely validated datasets for testing. Because of the large variability mentioned above, we considered it necessary to validate AID on both a public dataset and a real-world one, similar to what the authors of [41] did. Having made this premise, we turn to the description of AID. It consists of the following eight steps, depicted in Fig. 1:

- Data embedding;
- Preliminary assessment of cluster tendency;
- Dimensionality reduction;
- Clustering;
- Score function definition and hyperparameter setting;
- Cluster validation;
- Automatic labeling;
- Overall validation.

In the next subsections, we will explain each of these steps in detail, with a particular focus on automatic labeling. In fact, this step represents the most innovative part of our approach, as it operates at a high level of granularity using a comprehensive set of linguistic rules.

3.1. Data embedding

Several approaches have been proposed in the literature to construct numerical representations from natural language. In our scenario, the texts to be encoded consist of sentences of varying lengths (e.g., simple sentences, complex sentences, or short texts). From these, AID must derive the potential intents of a chatbot. To perform sentence encoding, AID uses two approaches already proposed in the literature and described in Section 2, namely the Universal Sentence Encoder (`USE`) [6] and the sentence transformers of Sentence BERT (SBERT) [36].

The `USE` model, developed by Google, is pre-trained on a wide range of web sources, including Wikipedia, news pages, Q&A sites, and discussion forums. This unsupervised training set is further enriched with labeled data from the Stanford Natural Language Inference (SNLI) corpus. `USE` is available in two variants, namely Transformers and Deep Averaging Networks (DANs). In the DAN model, sentence embeddings are produced from word embeddings and bi-grams, averaged together and processed by a feedforward deep neural network. The outputs returned by DAN models are 512-dimensional vectors.

As for sentence transformers, AID uses the "all" models, as they are specifically trained on all available training data (i.e., more than 1 billion training pairs). Moreover, the most recent and computationally sustainable versions of these models achieved the highest performance in terms of embeddings, as specified in [36]. Based on these considerations, the sentence transformers used in AID are:

- `all-mpnet-base-v2` and `all-distilroberta-v1`, which map sentences to a 768-dimensional vector space;
- `all-MiniLM-L12-v2` and `all-MiniLM-L6-v2`, which map sentences to a 384-dimensional vector space.

3.2. Preliminary assessment of cluster tendency

Once text embeddings have been generated, the next step in AID aims to assess the clustering tendency of resulting data. This is important to understand whether or not the inherent nature of available data makes them groupable in a meaningful way. Through such a check, we can avoid applying clustering to random data or data without a clear structure, which could lead to inaccurate or uninformative results. To perform this step, AID adopts the Hopkins statistic [23]. It is defined as follows: let X be a set of n data points, and W be a set of $m \ll n$ data points randomly sampled from X without replacement (clearly $W \subset X$). Let Y be a set of m data points uniformly randomly distributed in the same space as the previous data points. Y is not necessarily a subset of X. Let μ_i be the minimum distance (computed on the basis of some metric) between a point $y_i \in Y$ and its nearest neighbor in X. Let v_i be the minimum distance between a point $w_i \in W$ and its nearest neighbor in X. The Hopkins statistic is defined as:

$$H = \frac{\sum_{y_i \in Y} \mu_i^d}{\sum_{y_i \in Y} \mu_i^d + \sum_{w_i \in W} v_i^d} \tag{3.1}$$

Here, d represents the dimensionality of data. The ratio characterizing H is to compare the distribution of the randomly generated set Y and that of the sampled subset W of X. The value of H can belong to the real range $[0, 1]$. If it is close to 0.5, then we can conclude that the distributions are similar, which means that the data points of X tend to be randomly distributed. If it is low (e.g., less than 0.3), then the distribution of X should be considered regular. Finally, if it is high (e.g., greater than 0.7), then we can conclude that the data points of X have a high cluster tendency.

3.3. Dimensionality reduction

In the literature, several authors have raised the "curse of dimensionality" problem [1,34,2]. Indeed, in a high dimensionality space, data become excessively scattered, making traditional indexing techniques and algorithms either ineffective or inefficient. For this reason, reducing the number of dimensions is often necessary for an approach to work. Dimensionality reduction refers to the process of reducing the number of attributes in a dataset while maintaining as much of the original variability as possible. Dimensionality reduction methods fall into two macro-categories, namely:

- *Feature selection techniques*: they retain only the most important features without performing any transformation of them.
- *Feature extraction techniques*: they apply an appropriate transformation to the feature set by finding a new combination of them, thus projecting data points into a space with fewer dimensions. The set thus generated contains different values than the original ones.

As far as AID is concerned, we judged that, within its operational context, it is not appropriate to lose features. Accordingly, we focused on feature extraction techniques. In particular, we focused on two algorithms belonging to this kind of technique, namely t-SNE and UMAP.

t-SNE (t-distributed Stochastic Neighbor Embedding) [44] models a data point as a neighbor of another data point in both spaces, namely the original (high-dimension) and the final (low-dimension) ones. Thus, it aims to map high-dimensional data points onto a low-dimensional space while preserving pairwise similarities. To this end, it aims to minimize the divergence between the probability distributions in the high-dimensional and low-dimensional space. To reach this objective, it uses gradient descent until a stable state is reached.

UMAP (Uniform Manifold Approximation and Projection) [32] is based on Riemann geometry and algebraic topology. It uses several hyperparameters that control how dimensionality reduction is performed. Two particularly important hyperparameters are: *(i) n-neighbors*, which control how UMAP balances the local structures versus the global one; the lower the value of this parameter, the more UMAP focuses on local structure; *(ii) n-components*, which controls the dimensionality of the final embedded data after performing dimensionality reduction on the input data. As for our context, we observed from preliminary analyses that, compared with t-SNE, UMAP returns embeddings that better preserve the original structure of data. Furthermore, it features a shorter execution time and imposes no restrictions on the size of the input embeddings. Therefore, our choice fell on UMAP. Furthermore, we use UMAP not only for dimensionality reduction but also to visualize the clustering results in a bidimensional space.

3.4. Clustering

Our context requires a clustering algorithm that can handle clusters of arbitrary shapes, is tolerant of noisy data, and does not need the number of clusters to be specified in advance. The family of clustering algorithms that best meets all these criteria is the density-based one. Let us now examine some of the algorithms in this family to identify the most suitable one for our needs.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [13] assumes that clusters are dense regions of space separated by regions of lower density. It relies primarily on two parameters, namely: *(i) epsilon*, which represents the radius of the circle to be created around each point to verify density, and *(ii) minPoints*, which represents the minimum number of points required within the circle for the point at the center of the circle to be classified as a core. Our preliminary analysis revealed some weaknesses of DBSCAN in our context. In particular, it is not stable as the values of epsilon and minPoints vary. Furthermore, it is poorly suited with high-dimensional data because of the difficulties it encounters in defining density. Finally, it can be costly when computing nearest neighbors requires computing all pairwise proximities, which is common for high-dimensional data.

DENCLUE (DENsity CLUstEring) [20] is based on kernel density estimation, whose goal is to describe the distribution of data by a function, called kernel or influence function. Typically, kernel function is symmetric and its value decreases when the distance to the point increases. DENCLUE models the overall density of a set of points as the sum of the influence functions associated with each point. The resulting overall density function will have local peaks that can be used to define clusters in a natural way. In our preliminary analyses, DENCLUE has proven to be more sensitive and accurate than DBSCAN. However, it is computationally expensive. While grid-based techniques can mitigate this cost, they negatively impact the accuracy of density estimation. Additionally, DENCLUE is poorly suited for high-dimensional data, as well as data whose clusters are highly variable in density.

HDBSCAN (Hierarchical DBSCAN) [31] extends DBSCAN by converting it into a hierarchical clustering algorithm and then using a technique to perform flat clustering based on cluster stability. HDBSCAN is basically an implementation of DBSCAN that operates with variable values of epsilon. It requires only one input parameter, called *min_sample_size*, which represents the minimum cluster size. Unlike DBSCAN, HDBSCAN allows us to find clusters of varying density without having to choose a suitable distance threshold first. Furthermore, it allows us to obtain satisfactory results even with a large number of dimensions. All these reasons led us to choose HDBSCAN as the reference clustering algorithm in AID.

3.5. Definition of the score function and setting of hyperparameters

Once we have identified the most appropriate algorithms for dimensionality reduction and clustering, we obtain a pipeline with three hyperparameters (i.e., *n_neighbors* and *n_components* for UMAP, and *min_cluster_size* for HDBSCAN). This step aims to configure these hyperparameters, in order to obtain the best possible clustering outcome.

First, we need to define a score function to evaluate the clustering results. For this purpose, we exploit the attribute "*probabilities_*" of HDBSCAN. It is defined as the strength with which each sample is a member of the assigned cluster. Noise points are assigned a "*probabilities_*" value of 0, while the remaining data are assigned values for this attribute proportional to their degree of persistence within the cluster. This approach allows us to maximize the probability of assigning data points to their actual clusters while minimizing their misclassification as noise. Furthermore, to ensure a reasonable structure in the results while avoiding excessive fragmentation, it is essential to introduce a constraint on the number of resulting clusters. In case of violation of this constraint, a penalty $pnty$ is applied to the value of the score function.

Based on these premises, we can define the score function that we aim to minimize, which is given by the percentage of data points with a "*probabilities_*" value less than 0.05. Therefore, we have an optimization problem; in particular, we want to search for the values of the hyperparameters that minimize the score function while respecting the previously expressed constraint.

To solve this optimization problem, we chose to use the Bayesian optimization algorithm TPE (Tree-structured Parzen Estimators) [4]. It starts by making some assumptions about which are the best hyperparameters in the model and updates those assumptions as it learns how different hyperparameter values affect model performance. This way of proceeding allows it to make significant improvements over grid search and random search. In fact, instead of determining the best set of hyperparameters through trial and error, it gradually tries more combinations of hyperparameters that return good results and fewer combinations that lead to bad results. In fact, the name of this algorithm comes from two main ideas on which it is based, namely, the use of Parzen Estimation, to model hypotheses about the best hyperparameters, and the use of a tree-like data structure, to optimize the algorithm's runtime.

3.6. Cluster validation

To validate the obtained clusters, AID uses two types of metrics, namely extrinsic and intrinsic ones. The former compare the clustering results with an external ground truth, while the latter consider only the internal structure of clusters.

The extrinsic metrics used by AID are the following:

- *Adjusted Rank Index* (ARI): it measures how close the results of the clustering algorithm are to the ground truth by considering each pair of samples and checking whether the elements of the pair are assigned to the same cluster or different clusters by the algorithm being evaluated and the ground truth. ARI varies in the real interval $[-0.5, 1]$; the higher its value, the better the clustering algorithm.
- *Normalized Mutual Information* (NMI): it measures the mutual information between the results of the clustering algorithm and the ground truth. NMI varies in the real interval $[0, 1]$; the higher its value, the better the clustering algorithm.
- *Fowlkes-Mallows Index* (FMI): it computes the similarity between the results of the clustering algorithm and the ground truth according to the following formula [15]:

$$FMI = \frac{TP}{\sqrt{(TP + FP) \cdot (TP + FN)}} \tag{3.2}$$

  Here: *(i)* TP is the number of pairs of data points belonging to the same cluster in both the results of the clustering algorithms and the ground truth; *(ii)* FP is the number of pairs of data points belonging to the same cluster in the ground truth but not in the algorithm results; *(iii)* FN is the number of pairs of data points belonging to the same cluster in the algorithm results but not in the ground truth. FMI varies in the real interval $[0, 1]$; the higher its value, the better the clustering algorithm.
- *Homogeneity*: it indicates the extent to which the data points in each cluster, as returned by the algorithm into evaluation, belong to a class of the ground truth. It is defined as [39]:

$$Homogeneity = 1 - \frac{E(C|K)}{E(C)} \tag{3.3}$$

  Here: *(i)* $E(C)$ is the entropy of the class distribution; *(ii)* $E(C|K)$ is the conditional entropy of the class distribution given the assignment to the cluster. Homogeneity varies in the real interval $[0, 1]$; the higher its value, the better the clustering algorithm.
- *Completeness*: it indicates the extent to which the data points of each class of the ground truth are assigned to the same clusters. It is defined as [39]:

$$Completeness = 1 - \frac{E(K|C)}{E(K)} \tag{3.4}$$

  Here: *(i)* $E(K)$ is the entropy of the assignments of the data points to clusters performed by the algorithm into evaluation; *(ii)* $E(K|C)$ is the conditional entropy of the assignment of the data points to clusters given the class distribution in the ground truth. Completeness varies in the real interval $[0, 1]$; the higher its value, the better the clustering algorithm.
- *V-measure*: it is the harmonic mean of Homogeneity and Completeness [39]. Specifically, it is defined as:

$$V = 2 \cdot \frac{Homogeneity \cdot Completeness}{Homogeneity + Completeness} \tag{3.5}$$

  V-measure varies in the real interval $[0, 1]$; the higher its value, the better the clustering algorithm. It can be shown that V-measure is equivalent to Normalized Mutual Information.

The intrinsic metrics used by AID are the following:

- *Silhouette Score*: it indicates how similar the data points in the same cluster are to each other (cohesion) compared to those in other clusters (separation). It varies in the real interval [-1,1]; the higher its value, the better the clustering algorithm. This metric is particularly adequate when clusters tend to be globular in shape. Although this is not necessarily the case for clusters returned by density-based algorithms like HDBSCAN, we decided to use this metric, along with all the other more adequate ones, since it provides us with an additional perspective on the clusters obtained.
- *Density-Based Clustering Validation*: it evaluates density and shape of clusters taking into account noise [33]. It is the most appropriate intrinsic metric for evaluating potentially non-globular clusters, such as those returned by HDBSCAN. It varies in the real interval $[0, 1]$; the higher its value, the better the clustering algorithm.

3.7. Automatic labeling

The previous literature has shown that in conversational datasets with transactional interactions, such as those of interest for AID, utterances tend to feature an “action-object” type structure [26]. According to this paradigm, the “action” component identifies a command, a goal to be achieved or a task to be performed, while the “object” component represents the entity on which the “action” produces an effect. Drawing inspiration from [26], AID extracts the most frequent “action-object” pairs from each cluster and derives automatic labels from them. However, it introduces more detailed linguistic rules to improve the quality of results.

For the extraction of “action-object” pairs, AID uses Python’s spaCy library. In particular, among the available models for the English language, it uses `en_core_web_trf`. This model is based on RoBERTa [27] and has proven to be the most accurate for the two tasks necessary for our framework, namely:

- the PoS tagger, for identifying parts of speech;

- the dependency parser, trained on the Universal Dependencies (UD) corpus, for defining syntactic relationships.

In AID, rules focus on morphosyntax analysis, while also maintaining a consideration on semantic role theory. We describe them in the next subsections.

3.7.1. The "action" component

Regarding the "action" component, as in [26], we mainly consider the occurrences of the verb, i.e., the ultimate action category. Note that the tagging adopted by the PoS-tagger (PoS: *VERB*) does not include auxiliaries (labeled as PoS: *AUX*). In this way, we can exclude from the count those instances in which the verb only performs a grammatical function. To identify the most prominent instances, we primarily take into account verbs acting as the root of the syntactic tree (dep: *ROOT*). However, we add the following conditions to this general rule:

- *Modal and aspectual verbs*: in cases where the root verb falls into one of these categories, characterized by modal or aspectual meaning, we select the child verb in its place. This particular syntactic relationship is formally defined as "dep: *xcomp*" within the logic of UDs.
- *Coordinate structures*: coordination is treated asymmetrically by UDs, which analyze the second conjunct as a "child" of the main clause (dep: *conj)*. However, since this is not a real dependency relationship, we choose to select the verbs of both conjuncts whenever this type of relationship is present.
- *Content clauses*: if the head takes a noun clause, both serving as a subject or direct object (dep: *ccomp*, *csubj*, *csubjpass* or *acl*, depending on the clause's function), the selection falls on the verb of the complement, as this is more representative of the intent to be expressed. Instead, we exclude from the count the verbs of relative and adverbial clauses, since they only serve as modifiers of the main clause.
- *Adjectival predication*: if the predicate is expressed by an adjective, our selection falls on the latter (dep: *acomp*). This case, in fact, can be ascribed to the "action" category. Additionally, we observe that, unlike other copulative phrases (e.g., nominal predicate), adjectival predication often presents a formal resemblance to the passive voice.

In addition to these criteria for verb selection, some further rules are useful for a more accurate representation of the verbal phrase in its entirety to appear on the label. These are:

- *Phrasal verbs*: the PoS tagger and parser return a single tag for each text token, while always maintaining a 1:1 ratio between tokens and tags. However, the English language is rich in phrasal verbs, namely verb-adverb or verb-preposition compounds, acting as single semantic units. To represent this kind of verb, we consider the verbal head together with any particle that depends on it (dep: *prt*).
- *Negations*: whenever the verb (simple or phrasal) is in a negative form, the negation (dep: *neg*) - including contractions - and the verbal head are jointly selected. In verbs with adverbial modifiers (dep: *advmod*) we also consider the case where negation depends on the modifier and not directly on the verb. Finally, we also consider negation when it, although not direcly associated with the selected verb, depends on its head (or concatenation of heads), in relation to which it represents a phrasal complement.

3.7.2. The "object" component

Once the verb with the highest number of occurrences is identified among those meeting the conditions seen above, the most frequent "object" component is selected among all nouns (PoS: *NOUN* or *PROPN*) having one of the following relationships with that verb:

- *Direct object*: this is the case of the prototypical object, expressed by the relationship "dep: *dobj*" in the UD formalism.
- *Prepositional object*: like the direct object, it can often be taken as an argument by the verb (e.g., as a second argument by some prepositional verbs); it is expressed by the relationship "dep: *pobj*".
- *Subject of the intransitive verb* (dep: *nsubj* or *nsubjpass*): the choice to include this case in the list of objects, seemingly inconsistent from the syntax viewpoint, is justified by semantic role theory. Indeed, the first argument of intransitive verbs, especially when inaccusative, tends to play a less agentive role. This feature, not strongly evident in nominative-accusative languages, such as English, is evidenced cross-linguistically by the existence of ergative-absolutive or active-stative types of alignment.
- *Subject of adjectival predicate*: also when the predicate is expressed by an adjective phrase, the subject (dep: *nsubj*) does not play an agent role but represents the entity to which a characteristic or quality is attributed. On the basis of this reasoning, we consider assimilating such a subject to the "objects" of the predicate.

The indirect object (dep: *dative*) is not taken into account since its presence always implies the one of a direct object, which may be considered more prominent.

As with the "action" component, some rules are added to the "object" selection criteria to capture the noun phrase in its entirety. These include the following cases:

- *Phrasal nouns*: as with phrasal verbs (of which they often represent the nominalization), we jointly select the noun (PoS: *NOUN*) and the dependent particle (dep: *prt*).

Table 1
Semantic coherence scale.

Value	*Degree*	*Meaning*
0	Null	Sentences within the clusters represent completely separate intents and have no semantic affinity
1	Low	Some sentences in the clusters show some semantic similarity, but the intents are vague or poorly defined
2	Medium	Sentences in the clusters show reasonable semantic similarity and present generic but distinguishible intents
3	High	Sentences within the clusters exhibit a significant semantic similarity degree and reflect clearly related intents
4	Excellent	Sentences within the clusters are highly semantically consistent and show a strong and unambiguous link between them, representing unequivocal, well-delineated, and specific intents

- *Compound nouns*: in this case, we jointly select the noun serving as the head of the compound and the nominal modifier (PoS: *NOUN* or *PROPN*) having the *compound* dependence.
- *Adjectival modifiers*: while not being compounds in the strict sense, we select nouns jointly with their own adjectival modifiers (dep: *amod*) when the modifier-noun pair is more frequent than the single noun within the cluster.

3.7.3. Identification of additional characterizing words

The labeling approach proposed so far has some limitations; in fact, being based solely on syntactic parsing, it does not take into account aspects related to higher levels of language analysis, such as the presence of synonymy or coreference phenomena. Moreover, the "action-object" structure does not always fully take into account the complexity of the texts analyzed, as short as they are.

To overcome these limitations, AID concatenates two additional nouns, selected from the most frequent words within the cluster, to the "action-object" pairs. As a result of this operation, we obtain a second part of the label syntactically untethered from the first, which provides additional information useful for interpreting the intent. Specifically, this pair of words is chosen from tokens having *NOUN* or *PROPN* as a PoS and occurring more than once in the cluster. The same rules as in the "object" section regarding phrasal nouns, compound nouns, and adjectival modifiers apply to these tokens.

Since the purpose is to provide additional information that is not already captured by the "action-object" pair, we define two rules preventing redundancies between label components. Specifically:

- In case one of the most common words is already present in the first part of the label, we proceed to examine the most frequent words in the list, always ensuring that their occurrence within the cluster is greater than 1.
- In case of compounds or nouns with modifiers, if part of the compound already appears in the "object" component of the label, we select only the part of the compound that is not repetitive.

We close the description of the system of implemented rules with mentioning some expedients we adopted transversally along the labeling procedure. Specifically:

- The tokens evaluated for label construction were derived from their lemmatized forms; in fact, we considered that, in a frequency-based extraction system, the removal of inflectional morphemes is crucial to make the counting of occurrences more effective.
- Words separated by hyphens were counted as unique tokens (e.g., top up vs top-up); this allowed both the recognition of orthographic variability on certain phrasemes and the correct interpretation of codes containing punctuation marks within them.
- Punctuation was not taken into account.
- Words belonging to the "stop-word" list were generally excluded, except for those whose dependency was potentially useful for our analysis (dep: *neg*, *prt*, *amod*, *compound*, *xcomp*, *ccomp*, *dobj*, *pobj*, *csubj*, *csubjpass*, *conj*, *nsubj*, *nsubjpass*).
- Each selected item was considered only once within each sentence. The purpose of this expedient was to prevent a single sentence from unduly affecting the labeling of the whole cluster in the case where this sentence contains a large number of repetitions.

3.8. Overall validation

Once the labeling of clusters is completed, we can proceed with the overall evaluation of the results obtained. This task must take into account the semantic coherence, variability and label appropriateness criteria. It is carried out with the support of a human expert who is required to assign a score from 0 to 4 to each of these criteria. The meaning assigned to each criterion and the value of the corresponding scale are as follows:

- *Semantic coherence*: it assesses semantic similarity within clusters by looking at how strongly related the sentences in the clusters are in terms of their meaning. Ideally, sentences in the same clusters should have a high consistency degree, indicating that they belong to the same intent. The corresponding scale, along with its values and the meaning to be associated with each of them, is shown in Table 1.
- *Variability*: it assesses the variation within each cluster in terms of sentence structures, vocabulary, and wording used. Ideally, sentences within a cluster should capture different ways of expressing the same intent. The corresponding scale, along with its values and the meaning to be associated with each of them, is shown in Table 2.
- *Label appropriateness*: it assesses the quality of labeling assigned to each cluster in terms of relevance, interpretability, and specificity. The corresponding scale, along with its values and the meaning to be associated with each of them, is shown in Table 3.

Table 2
Variability scale.

Value	*Degree*	*Meaning*
0	Null	Sentences within the cluster are identical duplicates of the same phrase
1	Low	Sentences within the cluster share very similar templates with minor variations (regarding codes, names, and minor additions) from the default structure
2	Medium	Sentences within the cluster have moderate diversity in their structure, vocabulary, or wording
3	High	Sentences within the cluster show good variability in their structure, vocabulary, or wording, highlighting different ways of expressing the same intent
4	Excellent	Sentences within the cluster show a wide variability in their structure, vocabulary, or wording, fully representing the diversity of expression of the same intent

Table 3
Label appropriateness scale.

Value	*Degree*	*Meaning*
0	Null	The label is totally inappropriate and does not match the intent identified by the cluster
1	Low	The label is unrepresentative of the intent and is unclear, although not totally incorrect
2	Medium	The label is moderately clear and generally captures the essence of the intent identified by the cluster
3	High	The label is reasonably representative of the intent and offers useful insights regarding the content of the cluster
4	Excellent	The label is fully relevant, accurate, and easily interpretable

Since the proposed scales involve a degree of subjectivity, it is advisable that the evaluation of the previous three parameters is carried out by at least two evaluators. In this case, it is necessary to identify a metric that indicates the reliability and agreement of their judgments (inter-rate reliability). Among the metrics that can be used for this purpose, taking into account that the three parameters we have defined are ordinal, the Spearman's rank correlation coefficient [11] appears well suited.

In particular, let us consider one of the three criteria seen above (semantic coherence, variability and label appropriateness). Let u and v be two evaluators and let $\{u_1, u_2, \cdots, u_n\}$ and $\{v_1, v_2, \cdots, v_n\}$ be their ratings on the labels of the n clusters for that criterion. The Spearman's rank correlation coefficient is defined as follows:

$$\rho = 1 - \frac{6 \sum_{i=1}^{n} (u_i - v_i)^2}{(n^3 - n)} \tag{3.6}$$

The value of this coefficient ranges in the real interval $[-1, 1]$. The higher its value, the more the two evaluators are in agreement. We associate the Spearman's coefficient with a significance test. The null hypothesis of this test specifies that the two evaluators are not in agreement. Accordingly, we calculate the p-value and if this is less than the conventionally established threshold of 0.05 we can conclude that the null hypothesis can be rejected. In that case there is concordance between the two evaluators, and the mean of their evaluations for a cluster and a criterion can be taken as the final evaluation for that cluster and criterion. Otherwise, we have to conclude that the two evaluators are in disagreement. In that case, we proceed with representing to each evaluator the other's reasons; after that, we ask them to proceed with a new evaluation. After they return their new evaluation, we calculate its Spearman's rank correlation coefficient. If the corresponding p-value is less than 0.05, we can conclude that the two evaluators are in agreement and their new evaluation can be taken as the final one. If not, we ask a third evaluator to act as an arbiter between the two.

4. Experiments

In this section, we present the experimental campaign we conducted to evaluate AID. Specifically, in Subsection 4.1 we describe the datasets we used and highlight the differences between them. In Subsection 4.2, we illustrate the application of AID on these datasets and report the corresponding results. Finally, in Subsection 4.3 we propose a discussion on the results obtained.

4.1. Datasets

As pointed out in the Introduction, in order to carry out our tests, we used two datasets. The first of them is derived from the related literature. Instead, the second is taken from a real-world context; specifically, it concerns a set of questions made by the employees of a pharmaceutical company to an internal online helpdesk service.

4.1.1. Banking77

Introduced in [5], Banking77 provides a very fine-grained set of intents related to the banking domain. In Fig. 2, we report the 20 most frequent words in the dataset, representing some of the most recurring topics. The dataset consists of questions made online to the customer service, associated with a ground truth that categorizes them into 77 different intents. A full version of 13,083 and a sample version of 1,000 tickets are available to researchers. In order to make a more uniform comparison with the second dataset (see Section 4.1.2), we chose the sample version.

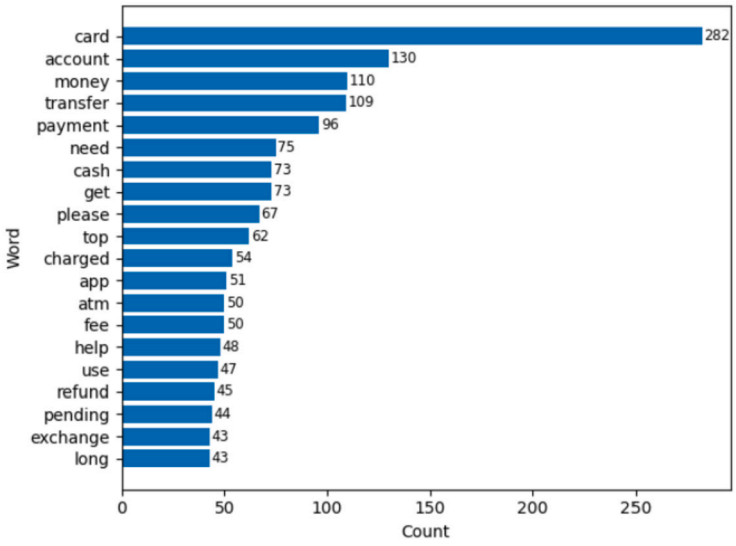

Fig. 2. Most frequent words occurring in the Banking77 dataset (stop words are excluded).

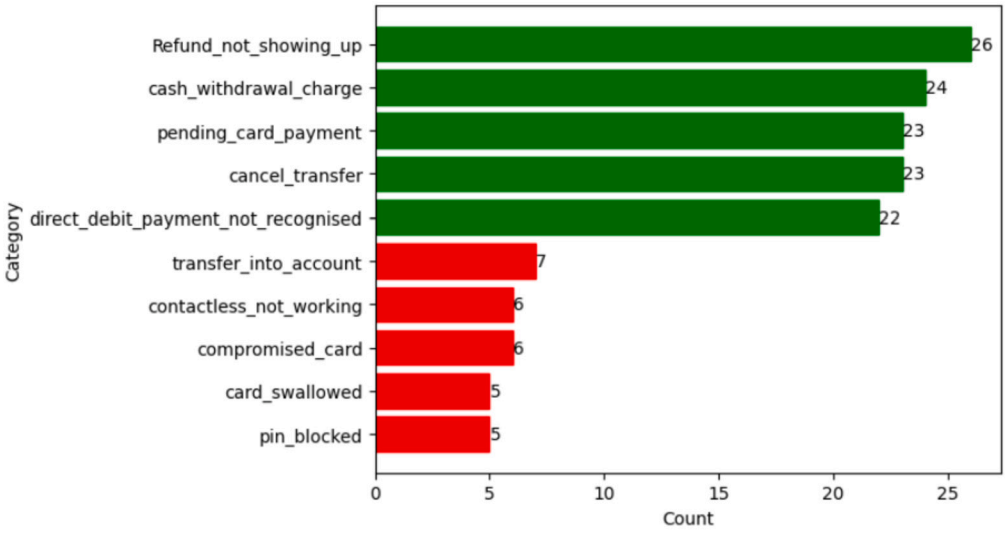

Fig. 3. Distribution of Banking77 categories - top 5 and bottom 5 ones by frequency.

Table 4
Ticket examples from Banking77 dataset.

Sentence	*Category*
Someone took my card without my permission.	lost_or_stolen_card
Is there a way I can check on the card on route to me?	card_arrival
Where are your cards delivered to?	order_physical_card

In Fig. 3, we report the five most frequent and five least frequent categories within the dataset. From the analysis of this figure we can observe a relatively uniform distribution of intents. Indeed, there is no marked disproportion among the categories with a high number of tickets, as well as among the ones with an extremely small number of tickets.

Other useful information that we need for comparing this dataset with the other one are: *(i) the average ticket length*, measured as the average number of characters in the corresponding question; *(ii) the Type-Token Ratio (TTR)*, an indicator of the lexical richness of the corpus, obtained by dividing its types (i.e., the total number of different words) by its tokens (i.e., the total number of words). In Banking77, the average ticket length is 59.2 characters, while the TTR is 0.13%. Therefore, the tickets are to be considered short. The relatively low TTR value reflects the sectorial nature of the language employed and suggests a probable repetitiveness of themes

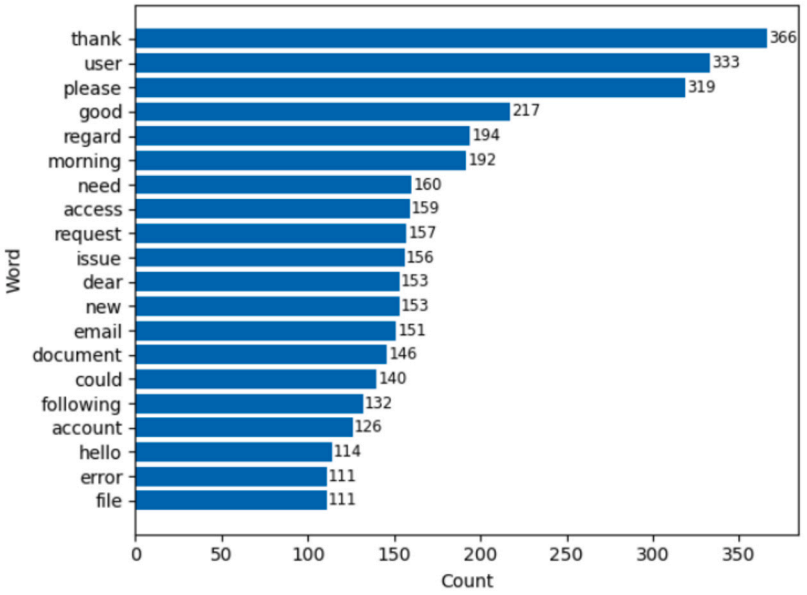

Fig. 4. Most frequent words occurring in the corporate dataset (stop words are excluded).

Table 5
Ticket examples from the corporate dataset.

Sentence
Hi, We have created a user in *[appName]*, but it has been generated incorrectly in the Active Directory: *[e-mail address]* should be *[e-mail address]*. Could you modify the email address? Thank you so much in advance. Regards, *[username]*
Please approve the mobile device management process for account *[e-mail address]*, device Xiaomi Mi10 Thank you, *[username]*
Good morning, I am unable to view the scanning folder on my PC. I request assistance to check its presence and, if it is not present, to add it. Thank you.
We need to eliminate version 1.1 of the documents *[document code]* and *[document code]* because these documents passed by a periodic review by mistake and we can't pass version 1.0 of these documents to Withdraw

among the submitted questions. In Table 4, we report some tickets extracted from the dataset and demonstrative of the characteristics just illustrated.

4.1.2. Corporate dataset

The corporate dataset consists of a set of questions asked by a company's employees to an internal online helpdesk service. It includes a total of 1,000 tickets and has no prior labeling. A preliminary quantitative analysis returns the idea of a more complex dataset on several levels. With an average of 272.3 characters per query and a TTR of 20%, the tickets are longer and more lexically varied than the Banking77 dataset. Fig. 4 illustrates the 20 most frequent tokens, offering a qualitative demonstration of potential complexity. Additionally, some examples of queries are presented in Table 5.

The examples given highlight the high occurrence of terms and expressions that are unnecessary for our purposes and could potentially mislead the clustering. These include generic words, greetings, thanks, and other introductory and closing formulas, including signatures of the message author. The frequency of numeric and alphanumeric codes, e-mail addresses and usernames is highly likely to be among the main causes of the increase in lexical variation of this dataset, witnessed by the high TTR. Also, due to the nature of the dataset, potential typos, as well as an improper use of punctuation and some text formatting choices (e.g., capital letters), must be taken into account.

To limit these problems, we performed a pre-processing activity that involved the following steps:

- *Removal of final signatures*, either in full or as initials. To identify the former we used the `en_core_web_trf` model of SpaCy. We took into account the last token (excluding punctuation) having *PROPN* as Part of Speech and being preceded by a punctuation mark other than colon. Along with this token, we selected all its *compound* dependencies, if any. To identify and remove initials ('letter.letter', or 'letter') we used RegEx patterns.
- *Removal of greeting and thanksgiving formulas*, identified through corpus-based research.
- *Removal via RegEx of isolated punctuation* resulting from typographical errors or the application of previous preprocessing steps.

Table 6
Example ticket from the corporate dataset, showing the importante of preserving stop words and digits.

Original Sentence	*Eventual cleaned sentence*
[appName], we are not able to update CN*[codeDigits]-[codeDigits]*	*[appName]* we able update cn

Table 7
Values of the Hopkins statistic obtained when it was applied on the Banking77 dataset.

Model	*Hopkins statistic*
`USE`	0.674
`all-mpnet-base-v2`	0.742
`all-distilroberta-v1`	0.735
`all-MiniLM-L12-v2`	0.747
`all-MiniLM-L6-v2`	0.744

Table 8
Best hyperparameter values, along with the corresponding score and number of clusters, obtained for all models adopted in AID when they are applied on the Banking77 dataset.

Model	*n_neighbors*	*n_components*	*min_cluster_size*	*Score*	*Number of clusters*
`USE`	3	10	5	0.077	57
`all-mpnet-base-v2`	5	14	5	**0.046**	61
`all-distilroberta-v1`	5	5	5	0.090	75
`all-MiniLM-L12-v2`	7	6	6	0.087	56
`all-MiniLM-L6-v2`	6	11	6	0.070	52

- *Lowercasing* of all tokens.

Interestingly, compared to classical preprocessing approaches, we retained digits (very recurrent in document codes, IPs, e-mails, etc.), stop words, and the rest of sentence punctuation. These choices are intended to avoid overly aggressive preprocessing and to preserve the informativeness of the sentence. In fact, since we are working with advanced sentence encoders, which can interpret contextual information in the text, it is critical that we maintain consistent and complete text sequences. A clarifying example is provided in Table 6, to demonstrate the importance of maintaining these three elements.

4.2. Framework application and results

4.2.1. Banking77

After having performed the embedding of tickets, we calculated the Hopkins statistic, to get a measure of the cluster tendency of the dataset. The results obtained for both `USE`[1] (see Section 3.1) and each of the chosen embedding models are shown in Table 7. From the analysis of this table, we can see that the dataset is clusterable for all models of sentence transformers. Indeed, the value of the Hopkins statistic is greater than 0.70 for all our models.

Therefore, we can proceed with the next step of AID. First, we observe that the authors of Banking77 released a ground truth, useful to validate the clustering task. The ground truth categories are visually shown in Fig. 5 through the colors of their data points. Therefore, the clustering activity performed by AID will have to be compared with it. As mentioned in Sections 3.3 and 3.4, AID uses UMAP to reduce the sample dimensionality and HDBSCAN to perform clustering. In Section 3.5, we highlighted that an optimization task is required to define the hyperparameters of UMAP and HDBSCAN. To carry out this activity, AID uses TPE (Tree-structured Parzen Estimator) with the constraint on the minimum and maximum number of clusters to be returned by HDBSCAN. As seen above, Banking77 involves 77 intents; one can think of associating a cluster with each intent, as well as a cluster with two (or more) semantically similar intents. In principle, although it is very uncommon, one could have two or more clusters associated with one intent, if the latter is not semantically well defined. For this reason, we constrained TPE by specifying the minimum (resp., maximum) number of clusters to be 50 (resp., 80). Running TPE on each of the models seen above yields the values of the three hyperparameters *n_neighbors*, *n_components* and *min_cluste_size* shown in Table 8. The same table also reports the number of clusters obtained and the score value (recall that the lower the score and the better the corresponding model).

From the analysis of Table 8, we can observe that the model guaranteing the lowest score is `all-mpnet-base-v2`. In Fig. 6, we show the results of the corresponding clustering activity.

The availability of a ground truth allows both intrinsic and extrinsic metrics to be used to validate results. Both of these types of metrics agree in identifying `all-mpnet-base-v2` as the best model for clustering our data points, as shown in Tables 9 and 10. Interestingly, the results obtained from this model are better than those achieved by `USE` across all metrics. It is worth pointing out

[1] As pointed out above, we adopt `USE` as a benchmark against which to evaluate the performance of our model.

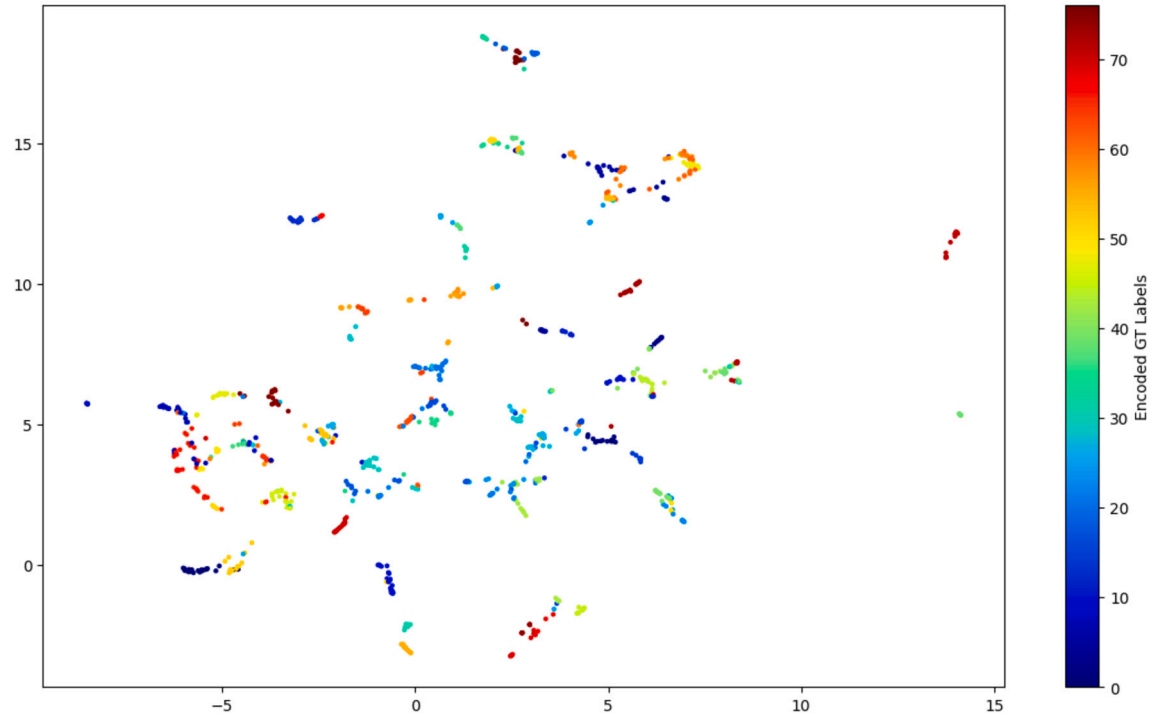

Fig. 5. The ground truth categories related to the Banking77 dataset.

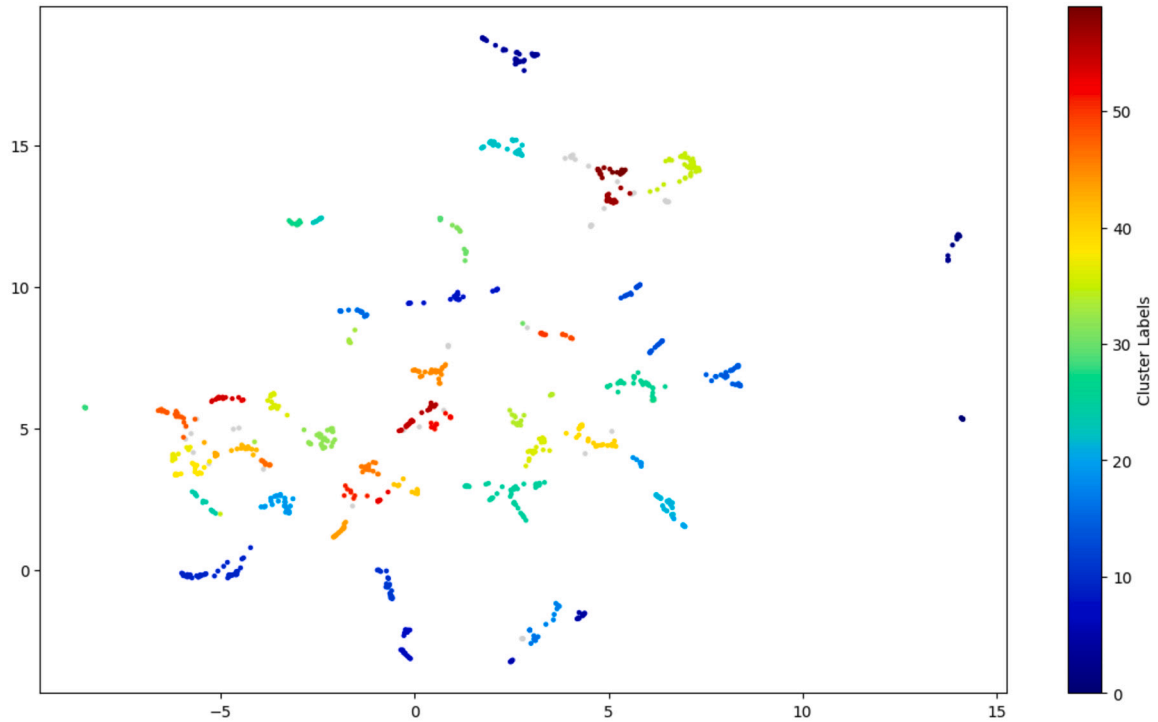

Fig. 6. Clustering results on the Banking77 dataset with the `all-mpnet-base-v2` model.

that, in terms of extrinsic metrics, the values achieved by our approach are better than the baseline values in unsupervised setting obtained in the literature on the same dataset, i.e., Banking77. In particular, Table 9 reports that our approach achieves a Normalized Mutual Information (NMI) value of 0.817 and an Adjusted Random Index (ARI) value of 0.484. These values are higher than those

Table 9
Values of the extrinsic metrics obtained for the Banking77 dataset.

Model	*Score*	*ARI*	*NMI*	*Fowlkes-Mallows*	*Homogeneity*	*Completeness*	*V-measure*
USE	0.077	0.257	0.700	0.296	0.656	0.752	0.700
all-mpnet-base-v2	**0.046**	**0.484**	**0.817**	**0.508**	**0.783**	**0.853**	**0.817**
all-distilroberta-v1	0.090	0.378	0.783	0.397	0.768	0.799	0.783
all-MiniLM-L12-v2	0.087	0.347	0.770	0.392	0.724	0.823	0.770
all-MiniLM-L6-v2	0.070	0.401	0.778	0.439	0.729	0.833	0.778

Table 10
Values of the intrinsic metrics obtained for the Banking77 dataset.

Model	*Score*	*Silhouette Score*	*DBCV*
USE	0.077	0.55	0.31
all-mpnet-base-v2	**0.046**	**0.64**	**0.42**
all-distilroberta-v1	0.090	0.56	0.22
all-MiniLM-L12-v2	0.087	0.55	0.23
all-MiniLM-L6-v2	0.070	0.59	0.31

Table 11
A cluster of the Banking77 dataset: its sentences, the labeling performed by AID and the one related to ground truth.

Sentence	*Automatic label*	*GT label*
Do I need to speak to a representative to change my pin?	change_pin	change_pin
Can I reset my PIN?	change_pin	change_pin
Do I need to go to a physical bank to change my PIN?	change_pin	change_pin
Can I change my PIN online?	change_pin	change_pin
Can my PIN changed remotely without visiting a bank?	change_pin	change_pin
Can I change my PIN at my local bank?	change_pin	change_pin
I'm traveling abroad but I've run into a situation where I need to change my PIN immediately. Can I do this from here?	change_pin	change_pin
What cash machines will let me change my PIN?	change_pin	change_pin

Table 12
Values of the Spearman's rank correlation coefficient and corresponding p-value for the Banking77 dataset.

Criteria	*Spearman index*	*p-value*
Semantic coherency	0.68	0.0010
Variability	0.87	0.0000
Label appropriateness	0.70	0.0006

obtained by the approaches of [48] (resp., [47]), which achieves an NMI value of 0.753 (resp., 0.678) and an ARI value of 0.433 (resp., 0.272).

After ascertaining the goodness of the clustering task through quantitative measures, in Table 11 we show an example cluster, including its sentences, automatic labeling generated by AID and ground truth labeling. From the analysis of this table, we observe a perfect alignment between the AID labeling and the one related to the ground truth.

As a final step of AID, we performed the human-expert assessment on the set of clusters obtained. For this purpose, we employed the three scales described in Section 3.8. Specifically, we asked two different experts to evaluate a sample of 20 clusters. Fig. 7 shows the scores assigned by each evaluator for the three parameters of semantic coherence, variability, and label appropriateness. From the analysis of this figure, we can see that evaluators were very satisfied with the results obtained by AID. In Table 12, we report the Spearman's rank correlation coefficient, along with the corresponding p-value. From the analysis of this table, we can see that the two evaluators showed a considerable agreement in their judgments. This can be seen from both the high values of the Spearman's coefficient and the very low values of p-value. This allows us to reject the null hypothesis of a discordance between the two evaluators.

4.2.2. Corporate dataset

In the case of corporate dataset, after performing the embedding of tickets (preprocessed as explained in Section 4.1.2), we evaluated the Hopkins statistic for USE and each model of AID. The results obtained are shown in Table 13. From the analysis of this table and the comparison with Table 7 we can observe that this dataset has a slightly lower clustering tendency than Banking77.

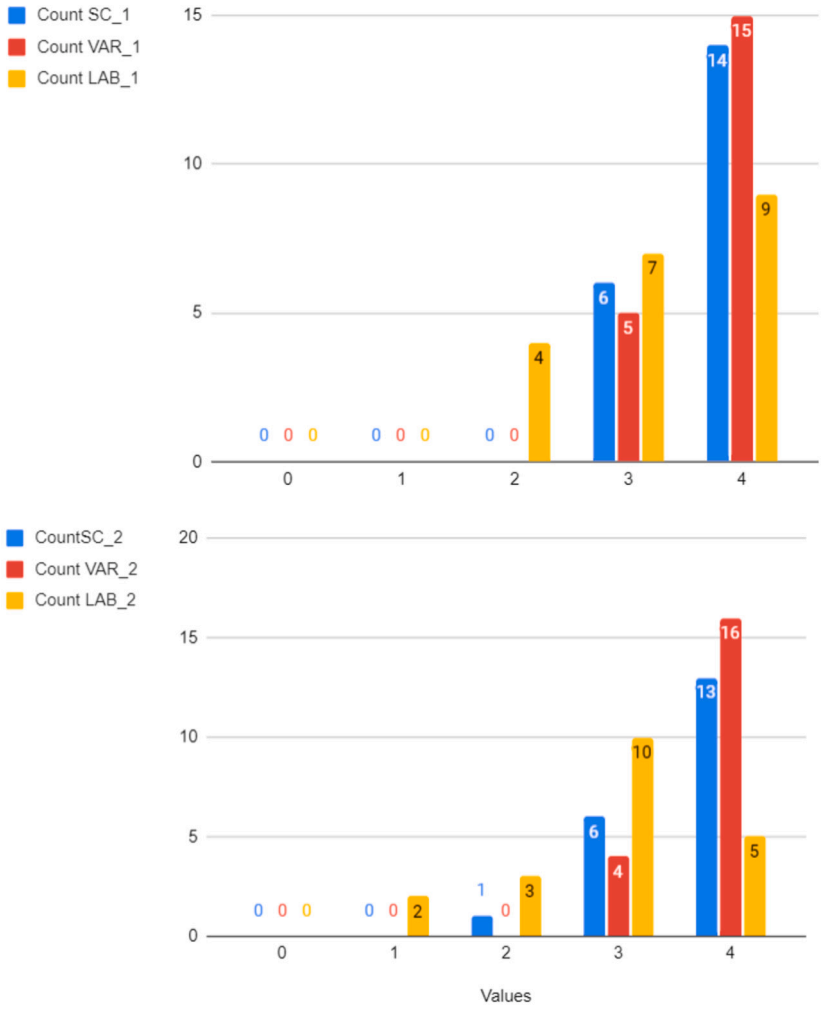

Fig. 7. Scores assigned by the two evaluators for semantic coherence, variability and label appropriateness - Banking77 dataset.

Table 13
Values of the Hopkins statistic obtained when it was applied on the corporate dataset.

Model	*Hopkins statistic*
`USE`	0.613
`all-mpnet-base-v2`	0.688
`all-distilroberta-v1`	0.680
`all-MiniLM-L12-v2`	0.678
`all-MiniLM-L6-v2`	0.677

However, the value obtained is sufficiently high, being just below the threshold of 0.70 for all models of AID and greater than 0.60 for `USE`. Therefore, we decided to proceed with the next steps of AID.

At this point, AID performs dataset dimensionality reduction, using UMAP, and data clustering, using HDBSCAN, with the simultaneous application of TPE for finding the best values of the corresponding hyperparameters. Since we do not have a ground truth in this case, it is much more difficult to define the constraints on the minimum and maximum number of clusters. For this reason, we chose a minimum number of clusters equal to 20 and a maximum number of clusters equal to 300. The reason behind this choice was to have a range of values as wide as possible, while avoiding overgeneralization (leading to uninformative clusters) or overly fine granularity (resulting in excessive templatization). The best score, the combination of hyperparameter values returning it and the number of final clusters for each embedding model considered by AID are shown in Table 14. From the analysis of this table, we can see that the best embedding model for the corporate dataset is `all-distilroberta-v1`. The clustering result obtained with this model and HDBSCAN is shown in Fig. 8.

Table 14
Best hyperparameter values, along with the corresponding score and number of clusters, obtained for all models adopted in AID when they are applied on the corporate dataset.

Model	*n_neighbors*	*n_components*	*min_cluster_size*	*Score*	*Number of clusters*
`USE`	3	6	2	0.085	167
`all-mpnet-base-v2`	3	11	2	0.085	166
`all-distilroberta-v1`	3	12	2	**0.058**	163
`all-MiniLM-L12-v2`	3	15	5	0.136	55
`all-MiniLM-L6-v2`	4	12	2	0.104	146

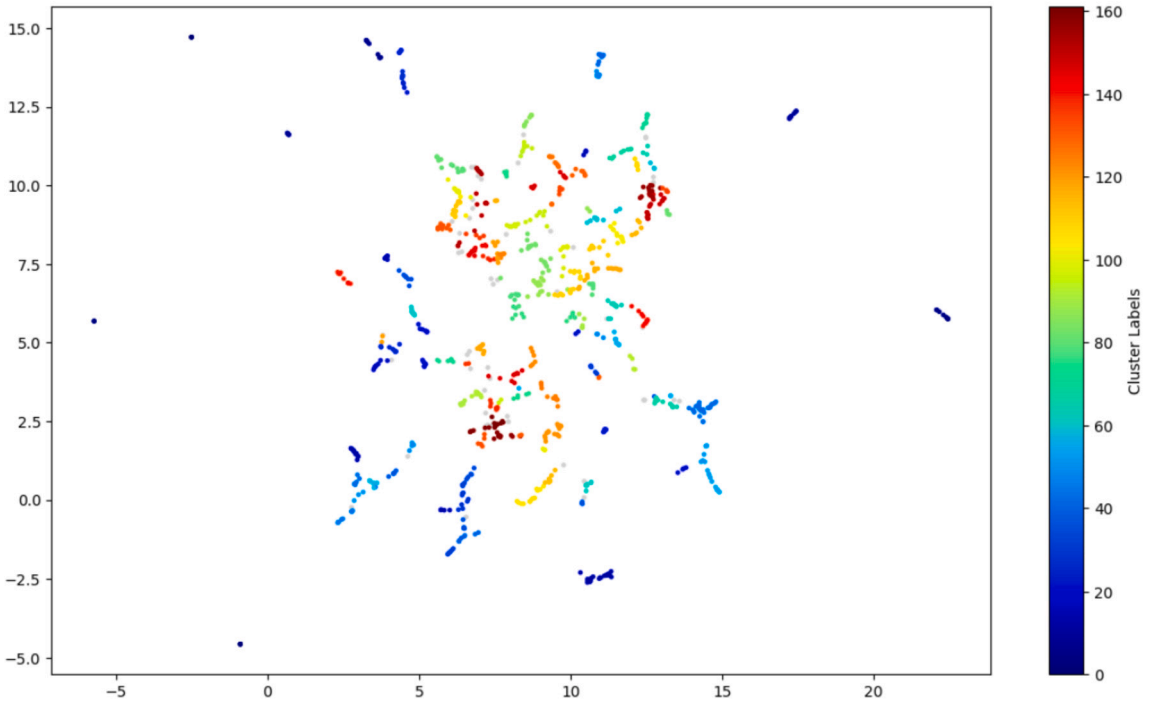

Fig. 8. Clustering results on the corporate dataset with `all-distilroberta-v1` model.

Table 15
Values of the intrinsic measures obtained for the corporate dataset.

Model	*Score*	*Silhouette*	*DBCV*
`USE`	0.085	0.46	0.19
`all-mpnet-base-v2`	0.085	0.51	0.28
`all-distilroberta-v1`	**0.058**	**0.55**	**0.34**
`all-MiniLM-L12-v2`	0.136	0.44	0.17
`all-MiniLM-L6-v2`	0.104	0.46	0.17

Table 16
A cluster of the corporate dataset: its sentences and the labeling performed by AID.

Sentence	*Automatic label*
dr *[Name]* and dr *[Name]* could not visualize *[productName]* material in the spanish environment	not_visualize_environment_material
i can not visualize spain's environment material, in the attachment you can see all the codes i can not visualize.	not_visualize_environment_material

As for the statistical validation of clustering results, the absence of ground truth allows us to rely only on intrinsic metrics. The values of these metrics for the corporate dataset are shown in Table 15. They confirm that the model to be preferred in this case is `all-distilroberta-v1`, which has already been identified as the best one through score minimization.

Similar to Banking77, we report an example cluster in Table 16, along with the automatic labeling obtained through AID.

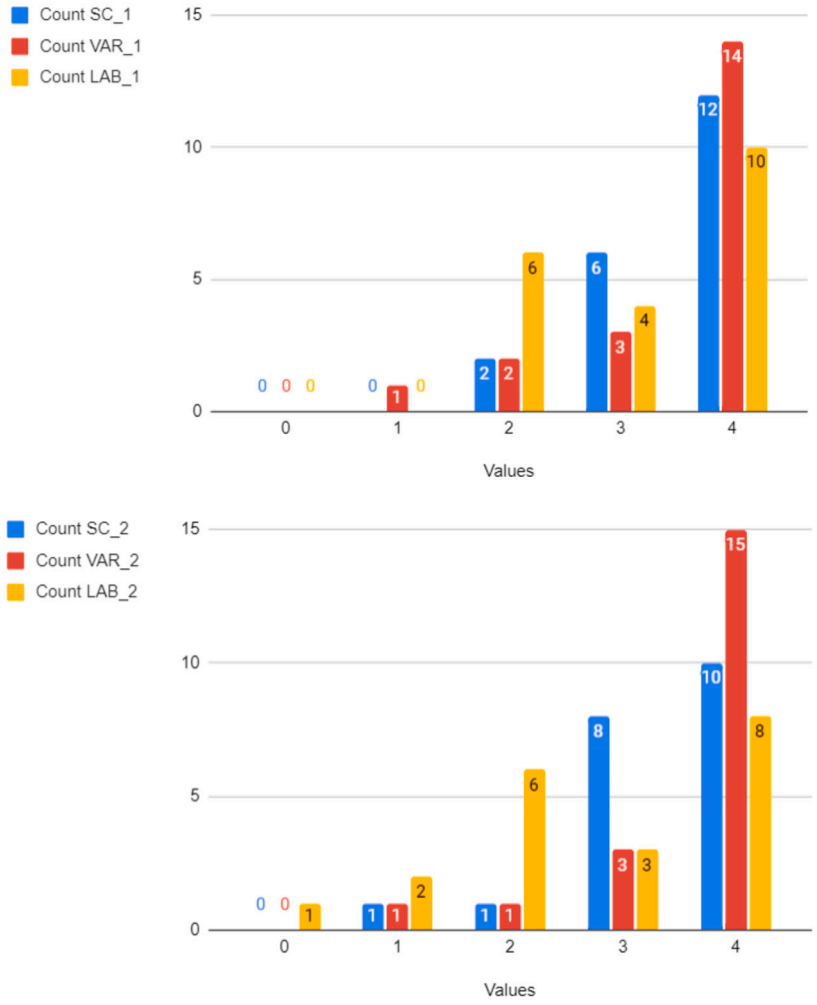

Fig. 9. Scores assigned by the two evaluators for semantic coherence, variability and label appropriateness - Corporate dataset.

Table 17
Values of the Spearman's rank correlation coefficient and corresponding p-value for the corporate dataset.

Criteria	*Spearman index*	*p-value*
Semantic coherency	0.70	0.0005
Variability	0.67	0.0012
Label appropriateness	0.62	0.0032

Once again, we enlisted the expertise of two evaluators to assess a sample of 20 clusters and the automatic labels defined by AID. In Fig. 9, we report the ratings of the evaluators, while in Table 17 we show the Spearman's rank correlation coefficient, along with the corresponding p-value.

From the analysis of Fig. 9 and Table 17 we can conclude that, even for the corporate dataset, evaluators are very satisfied with the results obtained by AID and that they essentially agree in this assessment.

4.3. Discussion

Having come to the end of our experiments, some remarks on the work done and the results obtained are in order. First, let us recall that the main objective of this work was to introduce a framework (which we called AID) for automatically detecting the intents of a conversational agent from an unlabeled dataset.

To validate the effectiveness of AID, we chose to first test it on a properly balanced public dataset (Banking77), consisting of fine-grained, well-categorized intents with ground truth labeling. The results obtained on this ideal dataset are very promising. In fact, the model detected as the one minimizing the score was confirmed as valid across all extrinsic and intrinsic metrics used in the

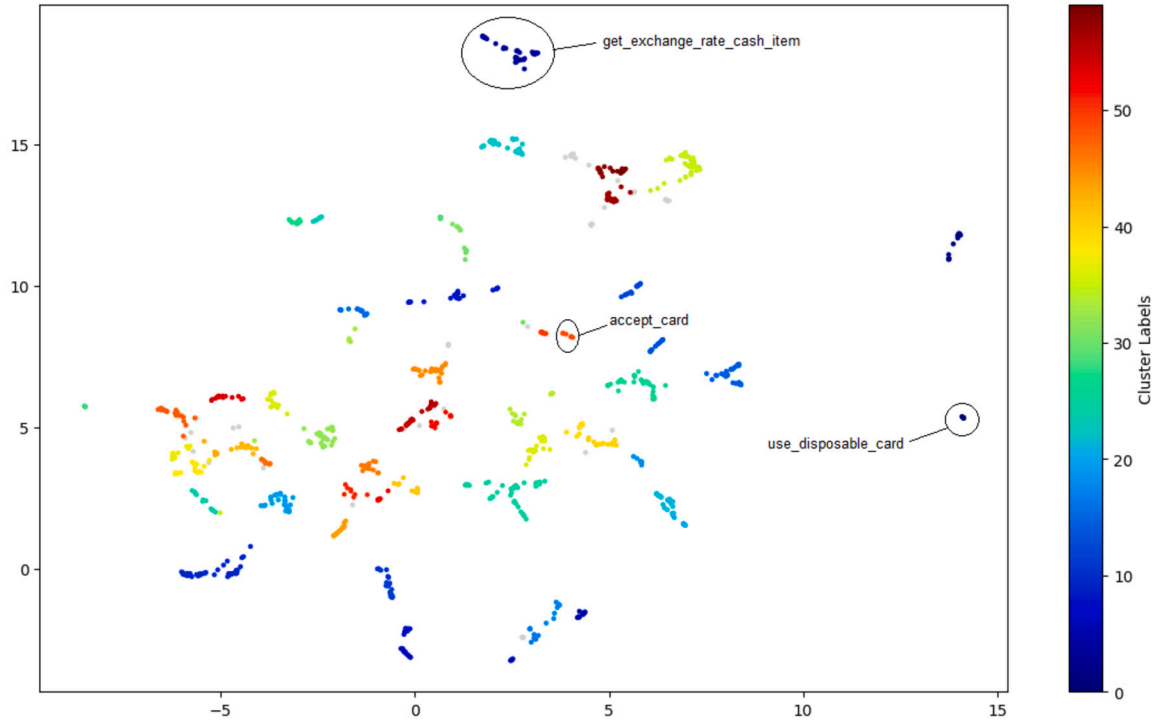

Fig. 10. Clusters of Banking77 considered in the qualitative investigation of the results returned by AID.

Table 18
A sample of the cluster `get_exchange_rate_cash_item`.

Sentence	*Automatic label*	*GT label*
I'm unhappy with the exchange rate your cash transactions.	get_exchange_rate_cash_item	wrong_exchange_rate_for_cash_withdrawal
Do you have the best exchange rate?	get_exchange_rate_cash_item	exchange_rate
My card payment has the wrong exchange rate	get_exchange_rate_cash_item	card_payment_wrong_exchange_rate

analysis. Regarding extrinsic metrics, we note a lower value of ARI than the other metrics considered. A possible explanation for this can be found in [38]. Here, the authors suggest to employ the ARI index preferably when the ground truth's clusters are large and similar to each other. They also say that NMI (i.e., the normalized version of AMI) is preferable when the clustering is unbalanced and, as in our case, clusters are small. Finally, although ARI is lower than the other extrinsic metrics, it still proves to be consistent with them, confirming that `all-mpnet-base-v2` is the model minimizing the score.

In addition to the results obtained from the previous quantitative analysis, some useful considerations can be derived from the qualitative investigation of clusters. This selection is based on a visual analysis of the results, identifying clusters that seem particularly interesting from the graph. For example, consider the topmost cluster in Fig. 10. In Table 18, we report a sample of it.[2] Our framework assigns to this cluster the label `get_exchange_rate_cash_item`. If we compare this scenario with the ground truth, we can see that this cluster includes (by generalizing them) three clusters of the ground truth whose labels are `wrong_exchange_rate_for_cash_withdrawal`, `exchange_rate` and `card_payment_wrong_exchange_rate`. Interestingly, these three clusters involve exchange rates, so it seems reasonable that they can be merged into a single cluster.

We perform the same analysis for a cluster positioned in the middle of Fig. 10 and shown in Table 19. AID assigns that cluster the label `accept_card`, which is interestingly very similar to the ground truth label `card_acceptance`. Afterward, we analyze the cluster on the far right of Fig. 10 and shown in Table 20. Similar to the previous case, the result of AID fully agrees with the ground truth. In fact, AID assigns it the label `use_disposable_card`, which is very similar to the label `get_disposable_virtual_card` related to the ground truth.

To conclude our discussion, let us now consider an example where AID does not perform very well. Specifically, we will examine a cluster with less satisfying results, which, precisely for this reason, offers valuable insights for future development of our approach. This cluster (positioned on the left in Fig. 10), a sample of which is shown in Table 21, is notable for its label, the second part of which (*pende) stands for the verb "to pend". Although the meaning of the cluster was correctly conveyed by the label (as evidenced

[2] We cannot report the whole cluster due to space limitations.

Table 19
The cluster `accept_card`.

Sentence	*Automatic label*	*GT label*
Is there a list of where I can use my card?	accept_card	card_acceptance
What businesses accept this card?	accept_card	card_acceptance
What retailers accept my card?	accept_card	card_acceptance
Is there restrictions on where I can use my card?	accept_card	card_acceptance
Where can my card be used?	accept_card	card_acceptance
Will my card work at all merchant locations?	accept_card	card_acceptance
Is there anywhere I can't use my card?	accept_card	card_acceptance

Table 20
The cluster `use_disposable_card`.

Sentence	*Automatic label*	*GT label*
What can you use the disposable cards for?	use_disposable_card	get_disposable_virtual_card
What is the function of the disposable cards?	use_disposable_card	get_disposable_virtual_card
Can you tell me what the disposable cards are used for?	use_disposable_card	get_disposable_virtual_card
How do I use the disposable cards?	use_disposable_card	get_disposable_virtual_card
What are the disposable cards used for?	use_disposable_card	get_disposable_virtual_card
How can disposable cards be used?	use_disposable_card	get_disposable_virtual_card
What are the disposable cards for?	use_disposable_card	get_disposable_virtual_card

Table 21
A sample of the cluster `have_pende_transfer`.

Sentence	*Automatic label*	*GT label*
How come I have pending transfers?	have_pende_transfer	pending_transfer
What is the reason why the transfer shows as pending?	have_pende_transfer	pending_transfer
Pending still shows on this transfer, why?	have_pende_transfer	pending_transfer
I can't figure out why a transfer is still pending?	have_pende_transfer	pending_transfer
I wanted to know why there is a transfer of mine pending.	have_pende_transfer	pending_transfer
My account says I have a pending transfer.	have_pende_transfer	pending_transfer

by the similarity to the ground truth label), this example demonstrates that AID is sometimes subject to errors by the lemmatizer. These errors are rather sporadic in predominantly isolating languages such as English, while they may become more common when trying to adapt AID's approach to morphologically more complex languages. This example and the reasoning behind it provide an important lesson for the future, as it prompts us to consider the importance of using the best language-specific lemmatization models from time to ensure consistently high quality results.

Therefore, we can conclude that the goal we had set with AID, namely capturing the ground truth or at most merging very similar clusters, has been achieved.

As for the corporate dataset, we note that, once again, the model minimizing the score also proves to be the best according to the intrinsic metrics. We also note that the values of the intrinsic metrics for this dataset are somewhat lower than those obtained for Banking77. In fact, data exhibit less inter-class separation with much concentration in the central area.

Finally, we observe that the best embedding model for Banking77 is `all-mpnet-base-v2`, while it is `all-distilroberta-v1` for the corporate dataset. The difference is likely due to the *Max Sequence Length* attribute, which represents the limit to the length of the sequences that can be passed to the model. In fact, in `all-mpnet-base-v2` this limit is set to 384, while in `all-distilroberta-v1` it is set to 512. This higher limit makes `all-distilroberta-v1` more suitable for processing the corporate dataset, which has a higher average number of characters per query. In any case, the purpose of our work was not to identify a unique embedding model valid for all datasets. Instead, we aimed to define a framework capable of discerning the best model for each scenario: in this regard, we can conclude that our goal has successfully been achieved.

5. Conclusion

In this paper, we presented AID (Automatic Intent Detector), a framework for automatic intent detection designed to support the development of transactional virtual assistants. AID is useful in the preliminary stages of conversational design, because it facilitates

the analysis of the large amounts of unstructured data from which the chatbot's knowledge base is developed. We presented the technical details of AID and outlined its main steps, which we determined through a study on the best solutions in the literature. Then, we described a comprehensive evaluation procedure that we defined to ensure result quality. Afterwards, we illustrated our experimental campaign conducted on two different datasets, namely Banking77 and a corporate dataset. This way of proceeding allowed us to demonstrate the validity of AID not only on ideal datasets, as it is generally the case in related work proposed in the literature, but also on a real-world corporate dataset, despite the potential criticalities of the latter. At this stage, we paid a great attention to the preprocessing phase, so that it is, on the one hand, minimal and noninvasive and, on the other hand, useful to make data more easily tractable.

In summary, the main contributions of our paper are as follows: *(i)* it introduces and defines the AID framework; *(ii)* it proposes an automatic method of labeling clusters in such a way that each label well represents the sentences contained in the corresponding cluster; this can be considered as the most important contribution, since it involves the creation of a detailed and sophisticated set of linguistic rules focusing on morphosyntactic analysis and incorporating semantic role theory; *(iii)* it proposes some metrics for validating AID and employs them to test it both on an ideal dataset widely used in the literature and a real one. As far as these objectives are concerned, the results obtained are satisfactory on both datasets. This stimulates us to explore new research directions to enhance the proposed framework.

A first possible development concerns the languages to which AID can be applied. The models employed so far are primarily tailored for processing English-language data. In the near future, we plan to extend support to other languages, evaluating the use of both multilingual and language-specific models. A second potential development concerns the evolution of AID toward the multimodality of the data processed. In fact, while the requests currently handled by AID are exclusively textual, in the future it is possible to consider the inclusion of a data normalization step, which would also enable the analysis of messages with audio and images. Finally, we observe that AID includes embedding models, as well as algorithms for dimensionality reduction, clustering, and Natural Language Processing. We believe that an unavoidable future effort is the continuous examination of new models and algorithms as they are proposed in the literature, in order to evaluate their possible integration into AID.

CRediT authorship contribution statement

Alessandra Ferrera: Visualization, Methodology, Formal analysis, Conceptualization. **Giulio Mezzotero:** Writing – original draft, Software, Project administration, Methodology. **Domenico Ursino:** Supervision, Investigation, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

References

[1] C.C. Aggarwal, A. Hinneburg, D.A. Keim, On the surprising behavior of distance metrics in high dimensional space, in: Database Theory—ICDT 2001: 8th International Conference London, UK, January 4–6, 2001 Proceedings 8, Springer, 2001, pp. 420–434.
[2] D. Angelov, Top2vec: distributed representations of topics, arXiv preprint, arXiv:2008.09470, 2020.
[3] M. Bali, S. Mohanty, S. Chatterjee, M. Sarma, R. Puravankara, Diabot: a predictive medical chatbot using ensemble learning, Int. J. Recent Trends Eng. Technol. 8 (2) (2019) 6334–6340.
[4] J. Bergstra, R. Bardenet, Y. Bengio, B. Kégl, Algorithms for hyper-parameter optimization, Adv. Neural Inf. Process. Syst. 24 (2011).
[5] I. Casanueva, T. Temčinas, D. Gerz, M. Henderson, I. Vulić, Efficient intent detection with dual sentence encoders, arXiv preprint, arXiv:2003.04807, 2020.
[6] D. Cer, Y. Yang, S.Y. Kong, N. Hua, N. Limtiaco, R.St. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, Y.H. Sung, B. Strope, R. Kurtzweil, Universal sentence encoder, arXiv preprint, arXiv:1803.11175, 2018.
[7] M. Chen, B. Jayakumar, M. Johnston, S.E. Mahmoodi, D. Pressel, Intent discovery for enterprise virtual assistants: applications of utterance embedding and clustering to intent mining, in: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track, 2022, pp. 197–208.
[8] Z. Chen, B. Liu, M. Hsu, M. Castellanos, R. Ghosh, Identifying intention posts in discussion forums, in: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2013, pp. 1041–1050.
[9] F. Colace, M. De Santo, M. Lombardi, F. Pascale, A. Pietrosanto, S. Lemma, Chatbot for e-learning: a case of study, Int. J. Mech. Eng. Robot. Res. 7 (5) (2018) 528–533.
[10] D. Comi, D. Christofidellis, P.F. Piazza, M. Manica, Zero-shot-bert-adapters: a zero-shot pipeline for unknown intent detection, in: Findings of the Association for Computational Linguistics: EMNLP 2023, 2023, pp. 650–663.
[11] A. De Raadt, M.J. Warrens, R.J. Bosker, H.A.L. Kiers, A comparison of reliability coefficients for ordinal rating scales, J. Classif. (2021) 1–25.
[12] P. Devine, Y.S. Koh, K. Blincoe, Evaluating unsupervised text embeddings on software user feedback, in: 2021 IEEE 29th International Requirements Engineering Conference Workshops (REW), IEEE, 2021, pp. 87–95.
[13] M. Ester, H.P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: KDD, vol. 96, 1996, pp. 226–231.
[14] E. Ferrara, Should chatgpt be biased? Challenges and risks of bias in large language models, arXiv preprint, arXiv:2304.03738, 2023.
[15] E.B. Fowlkes, C.L. Mallows, A method for comparing two hierarchical clusterings, J. Am. Stat. Assoc. 78 (383) (1983) 553–569.
[16] M. Grootendorst, Bertopic: neural topic modeling with a class-based tf-idf procedure, arXiv preprint, arXiv:2203.05794, 2022.
[17] H. Gu, Y. Xia, H. Xie, X. Shi, M. Shang, Robust and efficient algorithms for conversational contextual bandit, Inf. Sci. 657 (2024) 119993.

[18] H.B. Hashemi, A. Asiaee, R. Kraft, Query intent detection using convolutional neural networks, in: International Conference on Web Search and Data Mining, Workshop on Query Understanding, 2016, pp. 134–157.
[19] D. He, Y. Ren, A.M. Khattak, X. Liu, S. Tao, W. Gao, Automatic topic labeling using graph-based pre-trained neural embedding, Neurocomputing 463 (2021) 596–608.
[20] A. Hinneburg, D.A. Keim, An efficient approach to clustering in large multimedia databases with noise, in: Knowledge Discovery and Datamining (KDD'98), 1998, pp. 58–65.
[21] G. Hiremath, A. Hajare, P. Bhosale, R. Nanaware, K.S. Wagh, Chatbot for education system, Int. J. Adv. Res., Ideas Innov. Technol. 4 (3) (2018) 37–43.
[22] A.K. Kushwaha, A.K. Kar, Markbot–a language model-driven chatbot for interactive marketing in post-modern world, Inf. Syst. Front. (2021) 1–18.
[23] R.G. Lawson, P.C. Jurs, New index for clustering tendency and its application to chemical problems, J. Chem. Inf. Comput. Sci. 30 (1) (1990) 36–41.
[24] Q. Le, T. Mikolov, Distributed representations of sentences and documents, in: International Conference on Machine Learning, PMLR, 2014, pp. 1188–1196.
[25] B. Liu, I. Lane, Attention-based recurrent neural network models for joint intent detection and slot filling, arXiv preprint, arXiv:1609.01454, 2016.
[26] P. Liu, Y. Ning, K.K. Wu, K. Li, H. Meng, Open intent discovery through unsupervised semantic clustering and dependency parsing, arXiv preprint, arXiv: 2104.12114, 2021.
[27] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: a robustly optimized bert pretraining approach, arXiv preprint, arXiv:1907.11692, 2019.
[28] A. Lommatzsch, A next generation chatbot-framework for the public administration, in: Innovations for Community Services: 18th International Conference, Proceedings, I4CS 2018, Žilina, Slovakia, June 18-20, 2018, Springer, 2018, pp. 127–141.
[29] C. Magoo, M. Singh, An implemented review for intent creation using different clustering techniques, in: 2022 Fifth International Conference on Computational Intelligence and Communication Technologies (CCICT), IEEE, 2022, pp. 83–91.
[30] J.M.S. Martínez, S. Gorman, A. Nugent, A. Pal, Generating meaningful topic descriptions with sentence embeddings and lda, in: Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue, 2022, pp. 244–254.
[31] L. McInnes, J. Healy, S. Astels, hdbscan: hierarchical density based clustering, J. Open Sour. Softw. 2 (11) (2017) 205.
[32] L. McInnes, J. Healy, J. Melville, Umap: uniform manifold approximation and projection for dimension reduction, arXiv preprint, arXiv:1802.03426, 2018.
[33] D. Moulavi, P.A. Jaskowiak, R.J.G.B. Campello, A. Zimek, J. Sander, Density-based clustering validation, in: Proceedings of the 2014 SIAM International Conference on Data Mining, SIAM, 2014, pp. 839–847.
[34] A. Moura, P. Lima, F. Mendonça, S.S. Mostafa, F. Morgado-Dias, On the use of transformer-based models for intent detection using clustering algorithms, Appl. Sci. 13 (8) (2023) 5178.
[35] J. Nivre, D. Zeman, F. Ginter, F. Tyers, Universal dependencies, in: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Tutorial Abstracts, 2017.
[36] N. Reimers, I. Gurevych, Sentence-bert: sentence embeddings using Siamese bert-networks, arXiv preprint, arXiv:1908.10084, 2019.
[37] Y. Rizk, V. Isahagian, S. Boag, Y. Khazaeni, M. Unuvar, V. Muthusamy, R. Khalaf, A conversational digital assistant for intelligent process automation, in: Business Process Management: Blockchain and Robotic Process Automation Forum: BPM 2020 Blockchain and RPA Forum, Proceedings 18, Seville, Spain, September 13–18, 2020, Springer, 2020, pp. 85–100.
[38] S. Romano, N.X. Vinh, J. Bailey, K. Verspoor, Adjusting for chance clustering comparison measures, J. Mach. Learn. Res. 17 (1) (2016) 4635–4666.
[39] A. Rosenberg, J. Hirschberg, 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), 2007, pp. 410–420.
[40] N. Rosruen, T. Samanchuen, Chatbot utilization for medical consultant system, in: 2018 3rd Technology Innovation Management and Engineering Science International Conference (TIMES-iCON), IEEE, 2018, pp. 1–5.
[41] C. Shi, Q. Chen, L. Sha, S. Li, X. Sun, H. Wang, L. Zhang, Auto-dialabel: labeling dialogue data with unsupervised learning, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 684–689.
[42] D.Y.Y. Sim, C.K. Loo, Extensive assessment and evaluation methodologies on assistive social robots for modelling human–robot interaction–a review, Inf. Sci. 301 (2015) 305–344.
[43] A. Subakti, H. Murfi, N. Hariadi, The performance of bert as data representation of text clustering, J. Big Data 9 (1) (2022) 1–21.
[44] L. Van der Maaten, G. Hinton, Visualizing data using t-sne, J. Mach. Learn. Res. 9 (11) (2008).
[45] A. Xu, Z. Liu, Y. Guo, V. Sinha, R. Akkiraju, A new chatbot for customer service on social media, in: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, 2017, pp. 3506–3510.
[46] H. Zhang, W. Song, L. Liu, C. Du, X. Zhao, Query classification using convolutional neural networks, in: 2017 10th International Symposium on Computational Intelligence and Design (ISCID), vol. 2, IEEE, 2017, pp. 441–444.
[47] H. Zhang, H. Xu, T.E. Lin, R. Lyu, Discovering new intents with deep aligned clustering, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, 2021, pp. 14365–14373.
[48] H. Zhang, H. Xu, X. Wang, F. Long, K. Gao, A clustering framework for unsupervised and semi-supervised new intent discovery, IEEE Trans. Knowl. Data Eng. (2023).
[49] H. Zhang, Y. Zhang, L.M. Zhan, J. Chen, G. Shi, X.M. Wu, A. Lam, Effectiveness of pre-training for few-shot intent classification, arXiv preprint, arXiv:2109.05782, 2021.