Natural Language Processing (2024), 1–30
doi:10.1017/nlp.2024.45

ARTICLE

Learning and semiautomatic intention labeling for classification models: a COVID-19 dialog attendance study for chatbots

Valmir Oliveira dos Santos Júnior[1,2], Marcos Antonio de Oliveira[1,2], Lívia Almada Cruz[1] and Ticiana L. Coelho da Silva[1,2]

[1]Insight Data Science Lab - Federal University of Ceará, Brazil and [2]Graduate Program in Computer Science (PCOMP), Quixadá, Brazil
Corresponding author: Valmir Oliveira dos Santos Júnior; Email: valmir.oliveira@insightlab.ufc.br

(Received 15 December 2022; revised 7 August 2024; accepted 7 August 2024)

Abstract

It is increasingly common to use chatbots as an interface to services. One of the main components of a chatbot is the Natural Language Understanding (NLU) model, which is responsible for interpreting the text and extracting the intent and entities present in that text. It's possible to focus only on one of these tasks of NLU, such as intent classification. To train an NLU intent classification model, it's generally necessary to use a considerable amount of annotated data, where each sentence of the dataset receives a label indicating an intent. Performing manually labeling data is arduous and impracticable, depending on the data volume. Thus, an unsupervised machine learning technique, such as data clustering, could be applied to find and label patterns in the data. For this task, it is essential to have an effective vector embedding representation of texts that depicts the semantic information and helps the machine understand the context, intent, and other nuances of the entire text. This paper extensively evaluates different text embedding models for clustering and labeling. We also apply some operations to improve the dataset's quality, such as removing sentences and establishing various strategies for distance thresholds (cosine similarity) for the clusters' centroids. Then, we trained some intent classification Models with two different architectures, one built with the Rasa framework and the other with a neural network (NN) using the attendance text from the Coronavirus Platform Service of Ceará, Brazil. We also manually annotated a dataset to be used as validation data. We conducted a study on semiautomatic labeling, implemented through clustering and visual inspection, which introduced some labeling errors to the intent classification models. However, it would be unfeasible to annotate the entire dataset manually. Nevertheless, results of competitive accuracy were still achieved with the trained models.

Keywords: Intention model; chatbot; word embedding; sentence embedding; clustering

1. Introduction

Chatbots are software that tries to behave like humans in a conversation. The research in this field has advanced significantly, as evidenced by Kushwaha *et al.* (2021), Abdellatif *et al.* (2020), allowing chatbots not only to answer simple questions but also to perform complex tasks such as booking a hotel service and purchasing and selling crypto coins.

One of the main components of a chatbot is the Natural Language Understanding (NLU) model. NLU is responsible for text interpretation to make the conversation experience more

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

humanized and flexible. The NLU model combines machine learning (ML) and natural language processing (NLP) techniques to capture a user's intent and extract the entities related to the domain of the text posed by the user (Abdellatif *et al.* 2021).

An intent represents a mapping between what a user says and what action must be performed by the chatbot. Actions correspond to the chatbot's steps when the user triggers specific intentions. An entity is what or who is talked about on user input (Adamopoulou and Moussiades 2020). For example, consider the sentence "What are the places to visit in Fortaleza, Brazil?". The user intends to know about touristic places. The entity value is Fortaleza, Brazil.

However, as with any ML model, the NLU module training requires much-annotated data. Often, the data are annotated manually, which takes time and effort. This work proposes a methodology for semiautomatic annotation and learning of intention classification for chatbots. It is similar to what is presented by Peikari *et al.* (2018), where they first apply an unsupervised learning method (clustering) to find a pattern in data and then use these patterns (labels) to train a support vector machine (SVM) model to support them in taking decisions. Our use case is based on attendance at the COVID-19 dialog. The data comprise the dialogs of health professional advice carried out on the Platform of the Coronavirus Service[a] (PCS) in the state of Ceará, Brazil.

Applying unsupervised learning techniques to find patterns or links between the data is a way to aid the data annotation. In that method, the goal is to identify the data clusters most informative when taken as a whole and representing a class (Kassambara 2017). Text clustering is one of these strategies. We can apply a clustering technique to determine the intents represented by each cluster in a set of already-existing conversations. Then, the NLU intent classifier is trained using the intents as input.

Before performing text clustering, we must decide on the text representation given as input to the clustering algorithm. Due to its simplicity, a bag-of-words or bag-of-n-words (Harris 1954) is a typical form. However, these methods have drawbacks, including high dimensionality and sparsity. The capacity of pretrained word embeddings to capture a word's context inside a document and their semantic and syntactic similarities to other words has led to widespread use. However, word embeddings might not accurately capture meaning changes across sentences, even if they are just slight. Consider a sentence as "I do have a job," and another sentence as "I do not have a job." Despite the semantically opposing character of the phrases, word embeddings can produce cosine similarity vectors that are highly similar to these two phrases. Utilizing sentence embeddings, such as Yang *et al.* (2019); Cer *et al.* (2018); Le and Mikolov (2014), provides an alternative to this restriction.

Sentence embedding consists of encoding the entire sentence into embedding vectors. There are many pretrained sentence embeddings, including Doc2Vec (Le and Mikolov 2014), SBERT(Cer *et al.* 2018), and Universal Sentence Encoder (Reimers and Gurevych 2019). These embedding models accept the text as input and output fixed dimensional vector as the embedding representation of the full sentence. Such techniques try to capture the text's semantic information into the embedding vectors, which aids the machine in comprehending the context, intention, and other nuances of the entire text. The data made available for training has the most significant impact on the sentence embedding vectors. For optimal outcomes, it is crucial that the training set's sentences must be semantically related (Ham and Kim 2021).

This work extends the experiments presented by Dos Santos Júnior *et al.* (2021). We built several NLU models using the Rasa framework with different embedding models using the dataset described in this study. The dataset consists of dialogs collected from the PCS in Ceará, Brazil. The PCS platform features an online chat where individuals can connect with health professionals to receive recommendations about COVID-19.

[a] https://coronavirus.ceara.gov.br

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

There, the clustering process was simpler; instead, we used a small value for the K number clusters, K equals 10. However, in this work, we increase this limit, select the best value of K according to the clustering metrics, and adjust the sentences in the clusters according to different thresholds. So, the experiments in this new study address the issue of training an NLU intent model without annotated data. Given a dataset of dialogs, as input not annotated, we aim to propose an NLU intent model to create a chatbot. Besides these differences, we also study different embedding representation models, the labeling error added from the annotation processing, we improved the related work section, and we performed more experiments compared to our previous paper (Dos Santos Júnior *et al.* 2021). The main research questions that guide this study are as follows:

- **(RQ1)** From a huge conversation dataset, how do we label intentions using unsupervised learning for dialogs with short sentences without characterizing questions and answers throughout the conversation?
- **(RQ2)** How to create an NLU model for intent classification using the semiautomatically labeled data from **(RQ1)**?
- **(RQ3)** Could the embedding representation of texts used for the clustering step and labeling assist the training of an intent classifier?
- **(RQ4)** Since clustering is an unsupervised technique, does the clustering step add labeling error to the NLU intent classifier?

Our contributions are (i) the evaluation of different sentence embedding and word embedding strategies for the problem of discovering intentions in dialogs about COVID-19. We utilized a range of embedding methods, including GloVe, which offers static embeddings derived from word co-occurrence statistics. Additionally, we incorporated contextualized embeddings, primarily BERT-based, to capture contextual nuances in the texts and handle data sparsity challenges; (ii) an unsupervised proposal (k-means clustering-based) to deal with the need to annotate a conversation dataset with intention labels; (iii) a proposal of different approaches to deal with outliers in clustering processing for intentions labeling; (iv) investigation of the potential error included by the semiautomatic labeling; and (v) evaluation of how different embedding representations impact intent classification models. Regarding intent classification, we consider two architectures: a neural network (NN) based on a simple feed-forward and the Rasa NLU. The experimental dataset consists of 1,237 dialogs from PCS, collected between May 1, 2020, and May 6, 2020, with 26,754 sentences from patients and 26,992 sentences from health professionals, all annotated with their respective actors.

The remainder of this paper is organized as follows: Section 2 explains the preliminary concepts required to understand this work, and Section 3 presents some related works. Section 4 describes briefly the framework proposed for automatic intention classification. Section 5 discusses the data and methods to achieve our primary goals. Section 6 presents our experiments and their analysis. Section 7 discusses this work and its limitations. Finally, Section 8 summarizes this work and proposes future developments.

2. Background

This section provides an overview of the main concepts related to this paper.

2.1 Chatbots

Chatbots are artificial intelligence (AI) systems (Adamopoulou and Moussiades 2020) that can answer as an intelligent entity when conversing by text or voice. They can interact using more than one language using NLP techniques (Khanna *et al.* 2015). Besides imitating human conversation

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

and entertaining users, these systems can be used in applications such as education, healthcare, customer seervice, among others (Luo *et al.* 2022).

Chatbots can be classified considering different aspects related to **knowledge domain**, the **service offered**, the **goals**, the **entry processing way and answer generation methods**, the **human help,** and **construction methods** (Adamopoulou and Moussiades 2020).

As for the classification based on **knowledge domain**, they can be of **open domain**, where they can talk on general topics and answer suitably, instead of **closed domain**, where the focus is specific knowledge with much fewer generalization (Nimavat and Champaneria 2017).

Concerning the **service offered**, chatbots can be classified in **Interpersonal** and **Interagent**. **Interpersonal** chatbots are related to the communication domain, services such as helping with restaurant reservations, flight reservations, and Frequently Asked Questions (FAQ) bots. **Interpersonal**, generally make tasks from the user's domain, such as calendar management and user opinion storage, similar to what the human does. **Interagent** chatbots communicate among themselves to do some task (Nimavat and Champaneria 2017); they are predominant in areas such as the Internet of Things (IoT).

Another classification is related to the chatbot goals. An **informative** chatbot goals to provide information stored previously or available in some fixed data source, such as FAQ chatbots. A **conversational** chatbot goals fit the bots responsible for communicating with users as similar to a human as possible; generally, these are built using cross-questioning, evasion, and deference. Finally, the **task-based** chatbots act by making a well-defined task (such as a flight reservation or hotel reservation). Task-based chatbots can ask for information about the task at hand (Nimavat and Champaneria 2017).

Bots based on **entry processing way and answer generation methods** are classified in **rule-based**, **recovery-based**, and **generative**. The rule-based chooses an answer based on a fixed set of rules. The **recovery-based model** query and analyze resources made available through APIs, which offers more flexibility than the previous. A **generative** chatbot presents better answer generation than the others because it considers the current and prior messages, even though they are more onerous in the construction and training processes. They use ML and deep learning (DL) (Hien *et al.* 2018; Adamopoulou and Moussiades 2020).

Humans help chatbots by using human computation in at least one chatbot module. This approach can fill the slots caused by the limitations of the total automated bots. Even though human computation offers more flexibility concerning rule-based models and ML, they lose processing speed, reducing scalability (Kucherbaev, Bozzon, and Houben, 2018; Adamopoulou and Moussiades, 2020).

Development platforms for chatbots can be open source, such as Rasa,[b] allowing a greater variety of implementation aspects to be designed. On the opposite, closed platforms, such as IBM Watson,[c] work as a black box with less customization available, making challenging certain aspects of projects (Adamopoulou and Moussiades 2020).

Rasa is an open-source ML framework for automated text and voice-based conversations. It helps to understand messages, hold conversations, and connect to messaging channels and APIs. The Rasa NLU module works with a pipeline of components to train a model capable of extracting intents and entities from raw text using an annotated dataset as input. Rasa also provides tools for testing the performance of the NLU model. The pipeline can be customized to the model's necessities, making it possible to fine-tune the dataset. Pretrained word embeddings can be present in the pipeline, adding versatility to the trained model. Each component processes input and/or creates an output. The output of a component can be used by any other component that comes

[b] https://rasa.com/open-source/

[c] https://www.ibm.com/products/watson-assistant

after in the pipeline. Rasa provides many pretrained models for different languages, including BERT and GPT[d] (Rasa 2022).

For the intention classification model construction in this work, we have used the closed domain data from the Internet PCS in Ceará, Brazil. These are patients suspected of being infected by the COVID-19 illness. For that, we used two architectures: one uses the Rasa framework and the other uses NNs.

The intention classification model proposed can be used in an interpersonal task-oriented chatbot, allowing interaction with patients, asking questions about how the patients are feeling, intending to generate answers, and proposing actions to be taken by them, for example, looking for medical help when presenting specific symptoms related to COVID-19.

The next section discusses another relevant concept for this work: NNs.

2.2 Neural networks

Recently, NNs, specifically DL, have gained enormous space in the Speech Recognition research field, yielding better results than traditional methods (Nassif *et al.* 2019). DL consists of an ML algorithm whose input is multilayered models. NNs with different levels of nonlinear operations form ML algorithms that extract specific features and information from data (Nassif *et al.* 2019). A typical NN comprises layers of neurons that have activation functions and are connected through weights that are adjusted according to the feed of data and a back-propagation algorithm (Liu *et al.* 2017). In DL algorithms, layers are pretrained unsupervised and after being connected for supervised training and refinement (Liu *et al.* 2017).

Zhou *et al.* (2020) classify the neural NLP framework as: “modeling aimed at designing appropriate network structures for different tasks; learning aimed at optimizing the model parameters, and reasoning aimed at generating answers to unseen questions by manipulating existing knowledge with inference techniques.”

In what follows, we discuss the NLU and the intent classifier models.

2.3 Natural language understanding models

The NLU model identifies the user’s intents and extracts domain-specific entities. More specifically, intent summarizes the goal of the user input sentence and is used as a mapping between what a user says and what action must be performed by the chatbot. Actions correspond to the chatbot’s steps when the user triggers specific intentions. An entity is what or who is talked about on user input (Adamopoulou and Moussiades 2020).

One of the fundamental tasks in NLU is learning vector space representations of text. There are two popular approaches: multi-task learning and language model pretraining. These techniques are combined in a Multi-Task Deep Neural Network (MT-DNN) proposal. Human learning has inspired this approach in that they often apply the knowledge learned from previous tasks to help realize a new task (Liu *et al.* 2019b) and, as well, using tasks simultaneously can benefit from each other learned skills.

Gao *et al.* (2018) presents a survey with methods that demonstrate how pipelines of sequential tasks are applied to achieve NLU using language model pretraining. It is often necessary to fine-tune a pretrained model to specific NLU tasks for each task with additional task-specific layers using task-specific training data. Liu *et al.* (2019b) argues that multi-task learning and language model pretraining are complementary technologies, making possible their combination to improve the learning of text representations, increasing the performance of NLU tasks.

NLU is a preprocessing step for later modules in a chatbot system, and its performance interferes directly with the overall quality of the chatbot (Gao *et al.* 2018). Multiclass

[d] see https://huggingface.co/models for a complete list of available models

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

classification through neural approaches has become common in recent literature, and that technique is especially used for domain and intent classification tasks. For short sentences, where context is necessary to infer information, recurrent neural networks (RNNs) and convolutional neural networks (CNNs) are applied because they consider text before the current utterance (Lee and Dernoncourt 2016).

As for slot filling or entity identification, often sequence classification is used (Gao *et al.* 2018). In this approach, the classifier predicts semantic class labels for subsequences of the input utterance (Wang *et al.* 2005). Recurrent NNs are applied for this task, offering good results (Yao *et al.* 2013).

2.4 Intent classifier models

Determining the intent of a sentence uttered or typed during a conversation is crucial to the NLU task. Usually, a supervised learning model is used to classify intents, which are the goals of the uttered sentences. The following task is to fill slots representing semantic information that helps to fill the intent of the message (Weld *et al.* 2021). These two tasks can be performed during the NLU process sequentially in a pipeline or simultaneously in recent research. NLU is vital for a linguistic interface with humans. Technologies such as conversational agents, chatbots, the IoT, and virtual assistants, among others, need to do a good job of intent recognition, and science is rushing to do better in that sector. Word embedding and DL are among the recent technologies employed in the NLU process, achieving very promising results and adapting better to intelligent interfacing with humans, not only in text but also in voice and soon in videos and images (Liu *et al.* 2019a; Weld *et al.* 2021).

Some difficulties in intent detection are listed in Liu *et al.* (2019a): lack of data sources, irregularity of user expression, implicit intent detection, and multiple intent detection. The methods for intent detection can be traditional, such as rule-based template semantic recognition or classification algorithms based on statistical features. Common methods include Naive Bayes, SVM, and logistic regression. The current state-of-the-art methods include text representation via embedding, CNNs, RNNs, long short-term memory (LSTM) networks, gated recurrent unit (GRU), attention mechanism, and capsule networks. These DL models greatly improve detection performance (Liu *et al.* 2019a).

2.5 Clustering

Clustering is a class of unsupervised ML methods in which the basic problem consists of, given a set of data points, partitioning them into groups that are as similar as possible (Aggarwal and Reddy 2014).

There are many clustering algorithms in the literature. Among the clustering algorithms, k-means is one of the most popular. K-means works iteratively by finding the k centroids for k clusters and grouping every element from the dataset to the closest centroid. Initially, the algorithm randomly chose the k centroids (Vassilvitskii and Arthur 2006). Then, in each iteration, the centroids are computed as the average of all elements in a cluster. An important aspect is that the k-means algorithm is sensitive to outliers, although it performs well in computational time. Other popular strategies for clustering are the density-based algorithm DSBCAN (Ester *et al.* 1996) or the hierarchical clustering CLINK (Defays 1977). In this work, for the sake of simplicity, we use k-means. However, the approach in this paper fits with any clustering algorithm.

It is necessary to obtain its numeric representations for clustering in textual data. A possible use for embedding models is generating numeric vectors to represent textual data. The embedding models are discussed in the following.

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

2.6 Word embeddings

Vector space models transform the text of different lengths into a numeric fixed-length vector to be fed into downstream applications, such as similarity detection or ML models. Pretrained word embeddings have been widely used (Mikolov *et al.* 2013; Pennington *et al.* 2014; Akbik *et al.* 2019; Souza *et al.* 2020) due to their ability to capture the context of a word in a document, semantic, and syntactic similarity to other words.

Word2vec (Mikolov *et al.* 2013) is a framework for learning the word vectors by training a language model that predicts a word given the other words in a context. The main drawback is that it poorly utilizes the statistics of the corpus since the model is trained on a separate local context window instead of on global co-occurrence counts. Pennington *et al.* (2014) bypasses this problem and proposes a model that produces a word vector space. Pennington *et al.* (2014) trains the model on global word-word co-occurrence counts and efficiently uses statistics. FLAIR Akbik *et al.* (2019) abstracts away from specific engineering challenges that different types of word embeddings add to. FLAIR creates a unified interface for all word and sentence embeddings, as well as arbitrary combinations of embeddings.

BERTimbau (Souza *et al.* 2020) provides BERT models for Brazilian Portuguese. The models were evaluated on three NLP tasks: sentence textual similarity, recognizing textual entailment, and named entity recognition. BERTimbau improves the state-of-the-art. These tasks are done over multilingual BERT and previous monolingual approaches for Portuguese.

We can obtain document vectors from word embeddings by averaging all word vectors together. However, this procedure gives the same weight to important and unimportant words. Another limitation of representing text using word embeddings is each word would be embedded with the same vector regardless of the context. An extension of word embeddings is document or sentence embeddings to obtain the document vectors directly. From now on, we will consider sentence, document, and paragraph embedding to be the same.

2.7 Sentence embeddings

Sentence embedding represents sentences in a n-dimensional vector space such that semantically similar or semantically related words come together in the training method. Sentence embedding performs the representation of a sentence, which can have different representations of a word based on its context.

There are plenty of proposals for sentence embeddings as InferSent (Conneau *et al.* 2017), LaBSE (Feng *et al.* 2020), Universal Sentence Encoder Cer *et al.* (2018), Doc2Vec (Le and Mikolov 2014), among others. Universal Sentence Encoder proposes two different encoders. One uses the transformer architecture (Vaswani *et al.* 2017) and achieves the best performance. The attention mechanism computes context-aware representations of words in a sentence that takes into account both the ordering and identity of all the other words. The context-aware word representations are converted to a fixed-length sentence encoding vector by computing the element-wise sum of the representations at each word position (Cer *et al.* 2018). We refer the reader to (Galassi *et al.* 2021) for further details about the attention mechanism. The other proposed encoder is based on a deep averaging network (DAN) (Iyyer *et al.* 2015), whereby input word embeddings and bi-grams are averaged together and then passed through a feed-forward deep NN to produce sentence embeddings. Yang *et al.* (2019) extends (Cer *et al.* 2018) by proposing MUSE, a text embedding model for sixteen languages into a single semantic space using a multi-task trained dual encoder.

Similar to Word2Vec, Doc2Vec trains the paragraph vectors (or sentence embeddings) in the prediction task of the next word, given many contexts sampled from the paragraph. The paragraph and word vectors are concatenated to predict the next word in a context. LaBSE (Feng *et al.* 2020) is trained and optimized for multilingual sentence-level embeddings. It produces similar

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

representations exclusively for bilingual sentence pairs that are translations of each other. LaBSE employs a dual-encoder whereby source and target sentences are encoded separately using a shared BERT-based encoder and then feeding a combination function. The final layer [CLS] representations are taken as the sentence embedding for each input. The similarity between the source and target sentences is scored using cosine over the sentence embedding produced by the BERT encoders.

3. Related work

In this section, we discuss related studies to our problem. Due to the high demand for patient follow-ups, other groups have recently worked to develop chatbots related to COVID-19. Lei *et al.* (2021) train a NER model using scientific articles extracted from COVID-19 Open Research Dataset, CORD-19 (Wang *et al.* 2020). The papers' proposals often extract entities that are usable to identify symptoms in patients' written sentences. They use word clouds to find the most frequent symptoms cited in the articles, and the chatbot NLU model is used to build a knowledge graph that helps keep track of follow-ups from returning patients.

Fazzinga *et al.* (2021) apply natural language and argumentation graphs to build dialog systems that explain why a chatbot gave specific advice on COVID-19 vaccination. In Miner *et al.* (2020), the authors raise questions and problems that a chatbot could address during a pandemic like COVID-19. Initiatives such as Clara[e] from the CDC in the United States come to deal with the spread of conflicting information caused by lack of knowledge and fake news that ultimately can make dealing with the pandemic situation much more difficult.

Ouerhani *et al.* (2020) propose an intelligent and omnipresent propose an intelligent and omnipresent chatbot to help citizens understand the risks of COVID-19. The service is a mobile application on the web, structured in four independent components. The Information Understanding Module (IUM) converts user-typed data into structured data, intentions, and entities using NLP. The intentions are extracted using CRF, and the intentions are classified using the SVM algorithm. The Data Collector Module (DCM) collects nonconfidential information from the users to be used by the Action Generator Module (AGM). The AGM is responsible for generating the chatbot answers. To do that, it uses decision three algorithms based on a dataset built by the authors. The Depression Detector Module (DDM) detects anxiety in the text entering using a sentiment analyses model to help the AGM decide to send secure messages.

Due to the COVID-19 pandemic, many schools could not get prepared enough to demand virtual environment activities, Gaglo *et al.* (2021) have built a chatbot for aiding the tutoring of students from a high school in Senegal. The chatbot was built using the Rasa framework and integrated as a plugin from the learning virtual environment Moodle. The chatbot asks some questions to the students, tracing a profile for them, and with that, it could propose content for their studies. It is worth mentioning that teachers could consult all the dialog between the bot and the students. That allows analysis by the teachers over the bot tutoring, so they can measure the learning process quality and intervene with actions to improve it.

Klein *et al.* (2021) build a classifier using deep NNs based on a BERT model pretrained with COVID-19-related tweets. They have collected tweets from the Twitter Streaming application programming interface that mention keywords related to COVID-19. They have applied regular expressions to identify the tweets indicating if the user would have been exposed to COVID-19. The trained model can detect tweets that report potential cases of COVID-19.

In Judson *et al.* (2020), a chatbot is proposed to make the triage of employees during the change of shifts. The goal is to avoid hospital infection dissemination. Before the chatbot

[e] https://www.cdc.gov/coronavirus/2019-ncov/symptoms-testing/testing.html

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

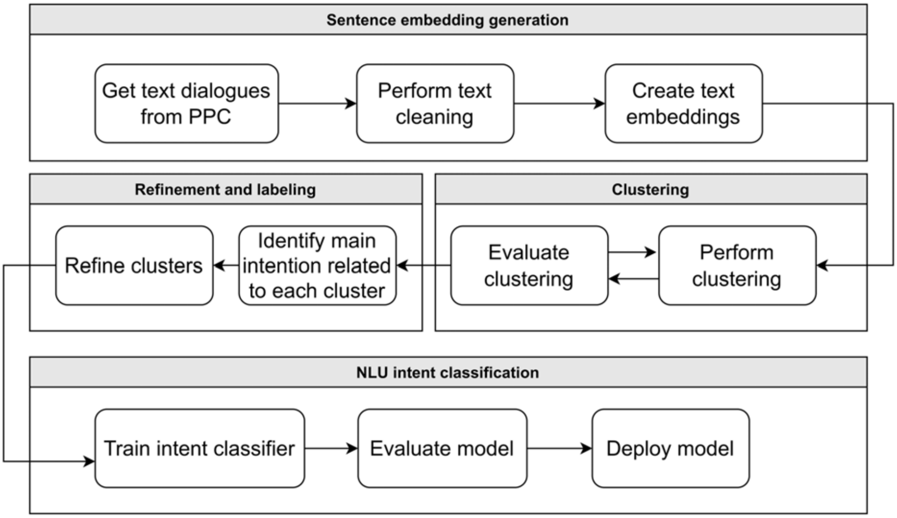


Figure 1. Pipeline to build NLU models and intent classifier models.

implantation, the collaborator needed to wait for 26 minutes to start attending. The solution allowed for diminished waiting time, and the triage was made from the collaborator's home, reducing the dissemination cases.

An ML model to characterize the current scenario on research related to COVID-19 is presented by Ebadi *et al.* (2021). They identify the latent topics and analyze the time evolution from the extracted research topics, the similarity, and the sentiment of the publications. They have used PubMed and ArXiv data obtained from January to May 2020.

Li *et al.* (2020) present the EmoCT (Emotion-Covid19-Tweet); for that, they have selected 1000 tweets randomly and classified each one into the following emotions: anger, acceptance, disgust, fear, happiness, sadness, surprise, and confidence. They used this dataset to train NLP classification models and applied the BERT embedding model to represent the tweets.

Aguiar *et al.* (2022) apply data augmentation to increase training data to response selection in chatbots based on multi-turn recovering. They apply the automatic translation of a massive dataset for multi-turn chatbots from English to Brazilian Portuguese, train a deep NN with the translated dataset, and tune the NN using a COVID-19-related dialogs dataset.

Peikari *et al.* (2018) argument data manually labeled is an arduous task. To minimize the effort on this problem, they first apply an unsupervised learning method (clustering) to find a pattern in a dataset of Pathology Images and then use these patterns (labels) to train the SVM model to make the classification. Compared with other state-of-the-art approaches, their observations showed promising results, showing that it is possible to use unsupervised learning methods to get labels for an unclassified dataset and use this dataset to train an ML model.

4. Pipeline for automatic intention classification

Before explaining the methods to achieve our research questions, we describe the process applied to build the NLU intent classifier model in this section. The pipeline used in this work, represented in Figure 1, consists of the general steps: i) sentence embedding generation, ii) clustering, iii) refinement, iv) intents labeling, and v) intention classifier learning.

One of the most common representations of document vocabulary is pretrained word embeddings. Another alternative is to encode the entire sentence into embedding vectors. This method is known as sentence embedding, and there are many pretrained sentence embeddings, as

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

mentioned before. The vector embedding generation step aims to generate vector representations for the patient's sentences. The vectors resulting from this step should capture syntactic and semantic information of sentences such that sentences expressing the same intention should have vectors close to each other in the new vector space.

The clustering phase goal is to group the sentence embedding vectors generated in the previous step to compose clusters of sentences concerning the same intention. We use separation and compactness metrics to set the hyperparameters used in the clustering algorithm, which obtained the best result. In general, a separation metric evaluates how well separated a cluster is compared to others, and a compactness metric evaluates how close the objects belonging to the same cluster are.

Following clustering, the clusters are visually inspected to determine the intention associated with each one. The sentences in the same group are labeled with the intention corresponding to the cluster. The same intention can be expressed in different ways. Thus, it is possible to recognize distinct clusters with the same intention.

Upon performing sentence labeling, we assume all sentences belonging to a cluster have the same intention. However, this hypothesis might not be valid. For this reason, before training the classifier, the refinement phase discards the less representative sentences from each cluster and outputs the most confident labeling for the intention classifier learning. This step is essential to remove some outliers not recognized by the k-means algorithm.

Finally, the sentences provided by the refinement step are used to train and validate the intention classifier.

5. Data and methods

This section provides the details about the dataset and methods applied in the pipeline described in the last section, from the data preprocessing to the building of the NLU intent classifier model.

5.1 Dataset

Dataset Description. The PCS has mechanisms for screening patients through interaction via chatbot. The service is an online chat where a chatbot performs the first interaction. Based on the patient's answer to some predetermined questions, the chatbot classifies the patient's condition according to criticality, which can be mild, moderate, or severe. After this first interaction, depending on the criticality classification, the patient is directed to teleassistance with a health professional. The interaction between the patient and health professional provides more details about the patient's conditions, including more specific symptoms, which can be physical or psychological. Table 1 shows examples of sentences between the Patient and the Attendant.

In the end, the patient evaluates the service. For this evaluation, the PCS requires the patient to answer the question "Are you pleased with the service?" and the response should be "Yes" or "No" and an evaluation score ranging from 0 to 10, where 0 is the lowest satisfaction rating and 10 the highest satisfaction rating. We aim to build a classification model to automatically recognize intentions using a not-labeled dialog dataset to identify patients' intentions while they report their health conditions. The intentions we are interested in here relate to the patient's diagnosis. Table 2 presents a fragment of a dialog showing this part where the PCS gets the evaluation and satisfaction level from the Patient.

The dataset used in the experimental evaluation is the set of dialogs between patients and health professionals with a positive evaluation collected from May 1, 2020, to May 6, 2020. We filtered only the dialogs assigned with a score of 10, which means the patient was satisfied with the attendance. After filtering, the dataset comprises 1, 237 dialogs, totaling 53, 746 sentences. The sentences from the dialogs are annotated with their actors (patients or health professionals), 26, 754 sentences are from the patients, and 26, 992 sentences are from the health professionals. As

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

Table 1. Dialog example

Actor	Sentence	Intention
Attendant	Hello goodnight. My name is . . .	greeting
Patient	I'm 03 days with fever and . . .	inform_symptoms
Attendant	Did you get to take the . . .	request_inform
Patient	Yes.	others
Attendant	Do you feel shortness of breath?	request_inform
Patient	. . . I took Dipirone. . ..	inform_medicine
Patient	. . . I think because of the cough	inform_symptoms
Attendant	right	others
Attendant	Do you feel shortness of breath. . .?	request_inform
Patient	only when I cough	inform_symptoms
Attendant	I advise contacting the . . .	inform_advice
. . .	. . .	. . .

Table 2. Fragment of a dialog showing the collection of the service evaluation performed by the Patient

Actor	Sentence
Bot	Thanks for getting in touch, Antonio! Be an ally
	in this fight, share this service with people and
	groups that may need our help.
	Together, we will win this fight against the new coronavirus!
	#StayAtHome
	Goodbye!
Bot	Have your queries been answered?
Patient	Yes
Bot	Please, before disconnecting, rate this service.
	Give a score from 1 to 10, where:
	1.Bad >>> 10. Very good!
Patient	10

in this work, the objective is to identify and learn patients' intentions. We used only the patient's sentences to build the NLU intent classifier.

Data Preparation. In the text cleaning, we first remove duplicate sentences from the dialogs. After, we select the dialogs from attendance that are highly well evaluated by patients, that is, attendance of health professionals with a score of 10. The filtering goal is to improve the quality of dialogs in the learning process. We also remove sentences represented by the following entities:

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

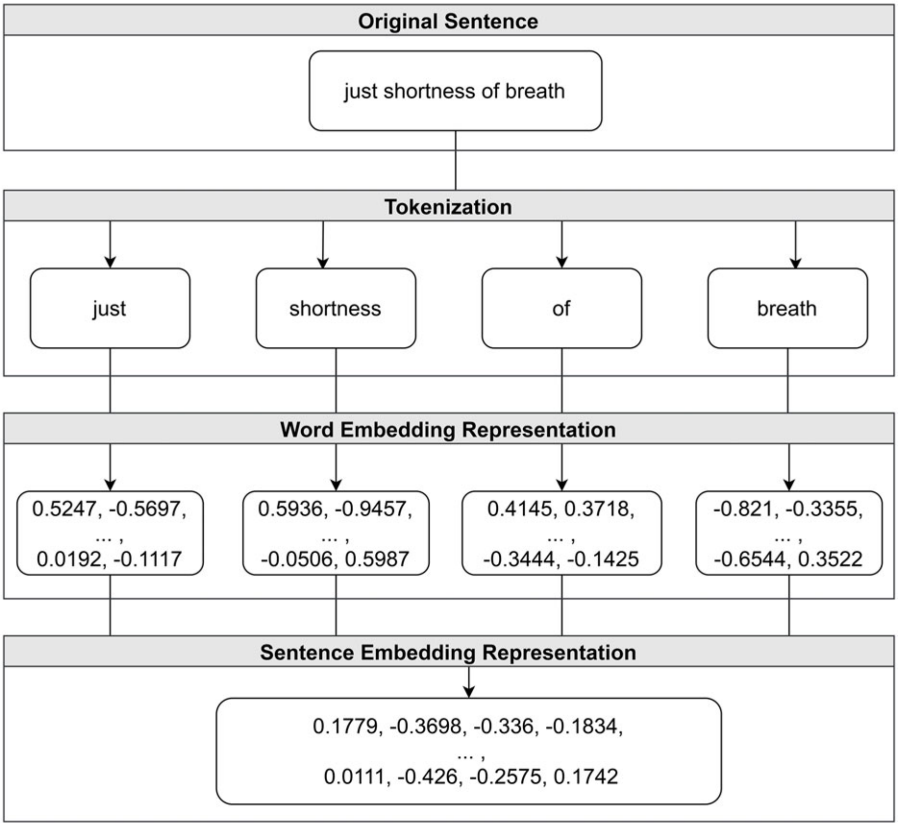


Figure 2. Pipeline to get embedding sentence representation using Glove.

ZIP code, Social Security Number (SSN), phone number, URLs, Emoticons, patient names, and places. We remove such entities manually to avoid the creation of clusters not directly related to a relevant intention. Moreover, it avoids using any personal information from users in the NLU intent model training process.

5.2 Methods

We now detail the methods concerning the pipeline steps explained in Section 4.

5.2.1 Sentence embedding generation

In this paper, we create sentence embeddings using a pretrained model from the state-of-art. In our proposal, we evaluate the sentence embedding models from MUSE (Yang *et al.* 2019) and LaBSE (Feng *et al.* 2020), and the word embedding models, FLAIR (Akbik *et al.* 2019), BERTimbau (Souza *et al.* 2020) and Glove (Pennington *et al.* 2014). We use the average vector of all word vectors as the sentence representation for the word embedding models.

For example, using the Glove embedding model to represent the sentence *"just shortness of breath,"* we have the following process shown in Figure 2. The first step is to tokenize the sentence. Each token found in the sentence is assigned a vector with a dimension of 300, where each item is a number (the word embedding representation); to simplify Figure 2 and make it readable, we show only the first two and the last two items of each generated vectors. In the last step, we apply an average calculation of all these vectors to get the sentence vector representation: a vector with the same dimensionality as the word embeddings. Again, we show only the first and last items of the sentence vector generated in this step. Other papers performed the same approach as in Dos Santos Júnior *et al.* (2021) and Wieting *et al.* (2016).

In the early stages of neural language models, foundational variants like Glove were the cornerstone of modern NLP by providing pretrained word features. The key advantage lies in these static models, usually employing a shallow NN architecture to perform computations between word vectors, facilitating efficient training. More recently, endeavors have emerged to acquire contextualized word representations through deep NNs, exemplified by BERT and BERT-based approaches. These embedding models stand at the forefront of the field, surpassing the static models in terms of state-of-the-art capabilities (Naseem *et al.* 2021).

5.2.2 Clustering

For this task, we use the well-known K-means clustering algorithm with cosine similarity. We variate the value of k and choose the best value of k based on the *Davies-Bouldin Score* (Davies and Bouldin 1979) and *Silhouette Score* (Rousseeuw 1987). Davies-Bouldin Score (DBS) is a separation metric given by the average similarity between a cluster and its most similar cluster. According to the DBS, the best clustering minimizes the average similarity and the lowest similarity value is 0. Thus, lower values of the DBS indicate better clustering. The idea behind using a separate metric is to reduce the overlap of intentions between clusters. The Silhouette Score (SS) indicates the ratio between cohesion and separation. It estimates the similarity between the object and its cluster compared to the similarity between the object and the other clusters. Since we experiment with multiple embedding models to represent text, each embedding approach may result in different "optimal" values of k.

5.2.3 Intents labeling

We use visual inspection to label the clusters and their assigned sentences. In general, visual inspection turns out to be a helpful tool whenever (1) different approaches produce clusters that have different semantics, (2) different sets of parameters yield clusters that perform well in terms of quality metrics but clearly show different characteristics, or (3) when ground truth is not available. We report that visual inspection was adopted in several past works as well (Ester *et al.* 1996; Han, Liu, and Omiecinski 2012; da Silva *et al.* 2020).

The visualization techniques used were t-Distributed Stochastic Neighbor Embedding (t-SNE) and word clouds. The t-SNE (Van der Maaten and Hinton 2008) approach allows visualizing high-dimensional data by converting each data point into a two-dimensional or three-dimensional space that displays the data structure at multiple scales. t-SNE allows visualizing the distribution of intentions and the clusters overlapping. The word clouds were applied to visualize the most frequent words in the clusters. Then, we label the sentences with their respective cluster intention to create the training and test sets for the NLU intent model. All in all, the training data consists of examples of user utterances categorized by one intent.

5.2.4 Clustering refinement

We filter the representative sentences before training the intention classifier and discard the outliers from the clusters. We define an upper bound for the permitted distance between the sentences in the cluster and its centroid, and then the sentences with a distance greater than the upper bound value are removed. We use the cosine distance between two sentences to calculate the distances between their embedding vectors.

Two strategies for the upper bounds are proposed. Let it be C a cluster and C_{Q_i} the ith quartile of the distances from the sentences in C to the centroid of C. The first strategy applies the removal of the outliers, and the upper bound is given by the Equations (1) and (2). The value *1.5* used in

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

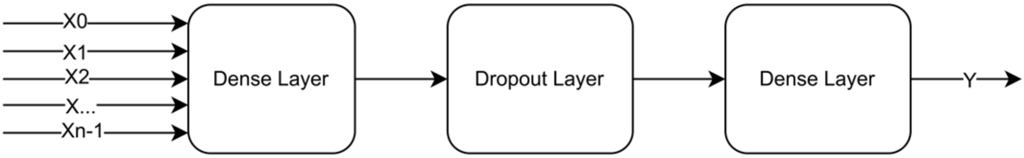


Figure 3. Intent classifier model architecture.

Equation (1) is a common value used to find the outliers of the dataset as it is explained in Hoaglin and Iglewicz (1987):

$$outliers\_upper\_bound(C) = C_{Q_3} + (1.5 * iqr) \tag{1}$$

$$iqr(C) = C_{Q_3} - C_{Q_1} \tag{2}$$

In the second strategy, the upper bound is the median of the distance (Equation (3)):

$$median\_upper\_bound(C) = C_{Q_2} \tag{3}$$

Besides the refinement of clusters, we also remove from the dataset all sentences belonging to the clusters where it is impossible to identify a precise intention, labeled with the intention *others*. We evaluate the different refinement strategies by measuring the clustering quality with and without the intent *others*.

5.2.5 Intent classification

For intent classification, we employ two architectures. The first is a Neural Network Intent Classifier (NN) (Figure 3) built using the embedding representation used for the clustering process. The NN classifier is based on a simple feed-forward NN. The sentence embedding vector is fed into the model and then processed by a full-connected layer. To reduce over-fitting, the input layer is linked to a dropout layer. The frequency rate used for the dropout layer was 0.1. A full-connected layer with *softmax* as an activation function produces the output. Finally, the number of intentions defined in the intent labeling step determines the output dimensionality. These models are trained using the *cross entropy* as loss function and Adam algorithm as optimizer.

The Rasa NLU is also applied as an intent classifier. The Rasa NLU is trained using the Rasa framework, which requires the specification of a Rasa pipeline comprising various components such as the tokenizer, feature extractor, and classifier architecture. These components sequence convert the input sentence into structured data, which is then passed to an ML model. For the Rasa pipeline, we use the spaCy module with the *SpacyNLP* component that receives a pretrained template from spaCy in the desired language (Portuguese in our case). In the pipeline applied, *SpacyTokenizer* is the tokenizer; and *SpacyFeaturizer*, *RegexFeaturizer*, and *CountVectorsFeaturizer* are the feature extractors that create the vector representation of messages to be given as input for the classifier. Note that using these components for feature extraction, the Rasa NLU intent classifier is agnostic to the embedding representation applied for clustering and labeling sentences in our pipeline. Finally, we apply the Dual Intent and Entity Transformer Classifier (*DIETClassifier*) as the classification model architecture. DIET is a multi-task architecture based on transformers that can predict intents and entities.

To assess the trained NLU intent model, we calculate the Matthews Correlation Coefficient (MCC) and the well-known standard metrics Precision, Recall, F1, and Accuracy score metrics.

The MCC considers true and false positives and negatives and is generally regarded as a balanced measure that can be used even if the classes are of very different sizes. In our case, we have multiclass labels, which have a minimum value ranging somewhere between −1 and 0 depending

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

on the number and distribution of ground true labels, and the maximum value is always + 1. The MCC can be defined as shown in Equation (4) (Matthews 1975):

$$\text{MCC} = \frac{c \cdot s - \sum_k^N p_k \cdot t_k}{\sqrt{(s^2 - \sum_k^N p_k^2) \cdot (s^2 - \sum_k^N t_k^2)}} \tag{4}$$

where t_k is the number of times intention I_k truly occurred, p_k is the number of times intention I_k was predicted, c is the number of samples correctly predicted, and s is the total number of samples.

In the experimental evaluation, we report the average precision, the average recall, the average f1, and the average accuracy along all intents, and MCC.

6. Experimental evaluation

This section presents the experimental evaluation that was carried out to assess the performance of our solution in answering our research questions.

6.1 Analysis of the annotation processing

The experiments performed and explained in this section remain to the research question **(RQ1)**: *From a huge conversation dataset, how to label intentions using unsupervised learning for dialogs with short sentences, without characterization of questions and answers throughout the conversation?*.

As we mentioned, our dialog dataset is not annotated regarding intention labels. One alternative is to perform a clustering strategy on our dataset according to the similarity between the patients' dialogs. Therefore, each cluster can represent an intention label associated with the texts assigned in that cluster.

Before clustering, we represent the sentences using embedding models. We evaluate the optimal number of clusters for each text embedding model by varying the number of clusters k between 2 and 100. In addition, we used the DBS and the SS, with cosine distance, as the metrics to choose the best value for k within this range.

Figure 4 shows the results for the DBS metric for all models. Remember, we aim to minimize the DBS value (the best value is zero (0)). Although all embedding models slightly perform the same, according to Figure 4, it's possible to notice that the embedding model with the best DBS value was Glove. Figure 5 shows the results for the SS metric for all models. It is important to remember that the best value of SS is one (1).

Table 3 shows the metrics, the values of k chosen according to the metrics, and the encoder type (the embedding approach). We can easily see that the values found for the SS are less than 1 and very close to 0, indicating an overlapping between clusters found by all the embedding models used in these experiments. This means the distance between a sentence S and the other sentences S' in the same cluster is almost the same as the distance between S and another sentence assigned to a different cluster. One reason might be because of the text vector representations have high dimensionality, that is, from around 500 to more than 4000. When we increase the dimensionality of the vectors, their pairwise distance tends to increase.

When you embed data into higher dimensions, it can affect the distance calculations between data points. In higher dimensions, the concept of distance becomes more complex due to phenomena like the curse of dimensionality Altman and Krzywinski (2018). This can impact how clusters are formed, and their centroids are calculated, affecting the DBS.

Higher-dimensional embeddings can potentially lead to increased inter-cluster and intra-cluster distances, which may influence the calculation of cluster separability and cohesion, which are factors in the DBS. Therefore, changing the dimensions of the embedding can alter

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

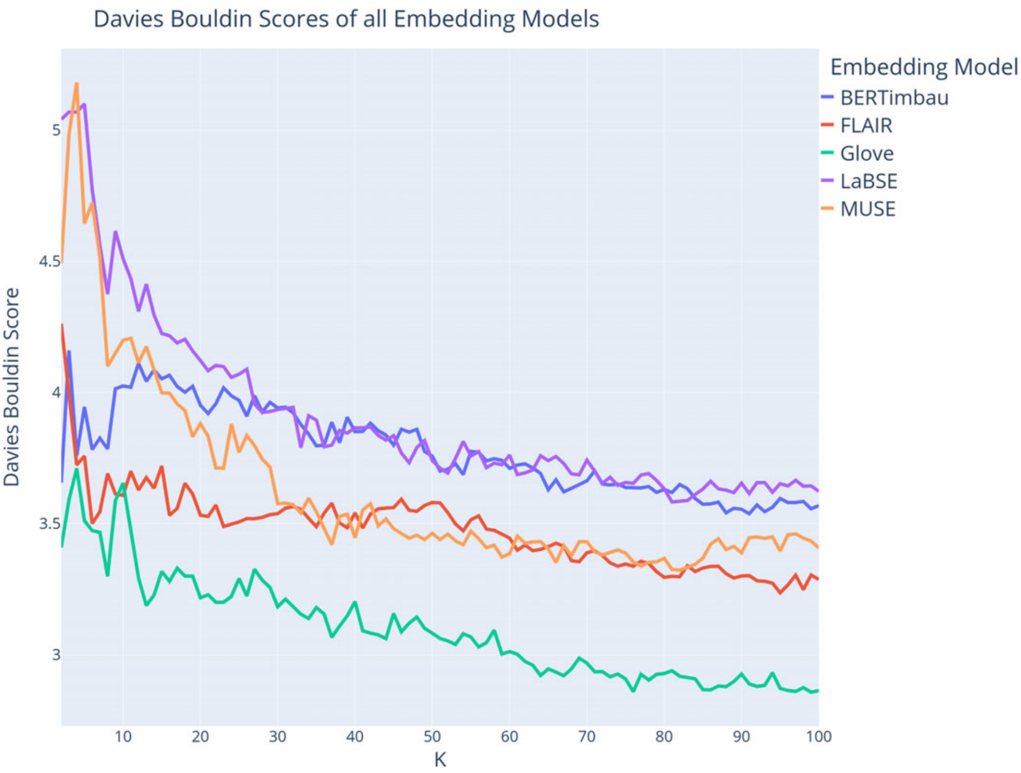


Figure 4. Davies Bouldin score of each embedding model.

the score because it changes the underlying geometry and distance calculations used in the clustering process.

So, for that reason, we chose to select the value of *k* based on the best value of DBS. In this case, the lower the DBS, the better the clusters. To further complement the evaluation of the clustering quality, we *visually inspect* some of the results produced by the clusters from the sentences embedded with each embedding model. For each embedding model, the best number of clusters (*k*) is almost 100, which means inspecting so many clusters visually is unfeasible. So, we visually inspected some of the generated clusters throughout t-SNE and word clouds. We presented these visualizations for the clusters from the embedding model representation, which achieved the best performance. From Table 3, Glove obtained better results for the DBS metric. However, it is important to note the good values of DBS for GloVe do not necessarily translate to superior semantic understanding. In contrast, a better DBS may imply a better semantic structure for semantic models such as BERT, BERTimbau, or MUSE.

Figure 6 shows examples of word clouds built from some groups of sentences obtained through clustering, performed using the Glove embeddings model. It is important to emphasize that each word cloud is formed only by the sentences of a particular cluster, which was labeled with a specific intention. Figure 6a shows a word cloud for one cluster representing a Greeting intention. Figure 6b shows a cluster, representing intent Inform Symptoms. Figure 6c shows a cluster, representing intent Inform Medicine. Figure 6d shows a cluster representing intent Request Inform. Figure 6e shows a cluster representing the intent of others.

The t-SNE visualization technique was used to present the distribution of sentences on the vector space. Figure 7 shows the distribution of the sentence vectors for the **ninety-nine** clusters generated using the Glove embedding model; the sentences with the same color belong to the same Intention. We can verify by Table 4, which shows some sentences from the glove clusters

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

Table 3. K value chosen for each embedding model representation

Embedding	Encoder	Dimensionality	K	DBS	SS
BERTimbau	Word	512	91	3.5375	0.0323
FLAIR	Word	768	95	3.2366	0.0393
Glove	Word	768	99	**2.8577**	0.0788
LaBSE	Sentence	300	81	3.5825	0.0688
MUSE	Sentence	4096	82	3.3227	**0.0928**

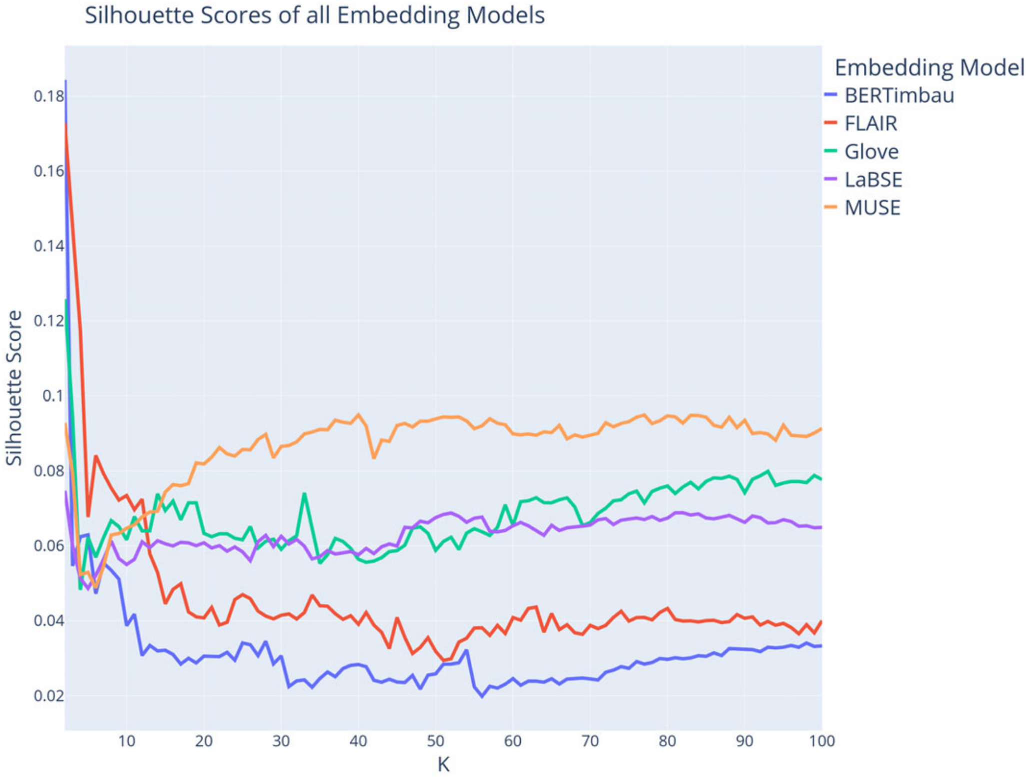
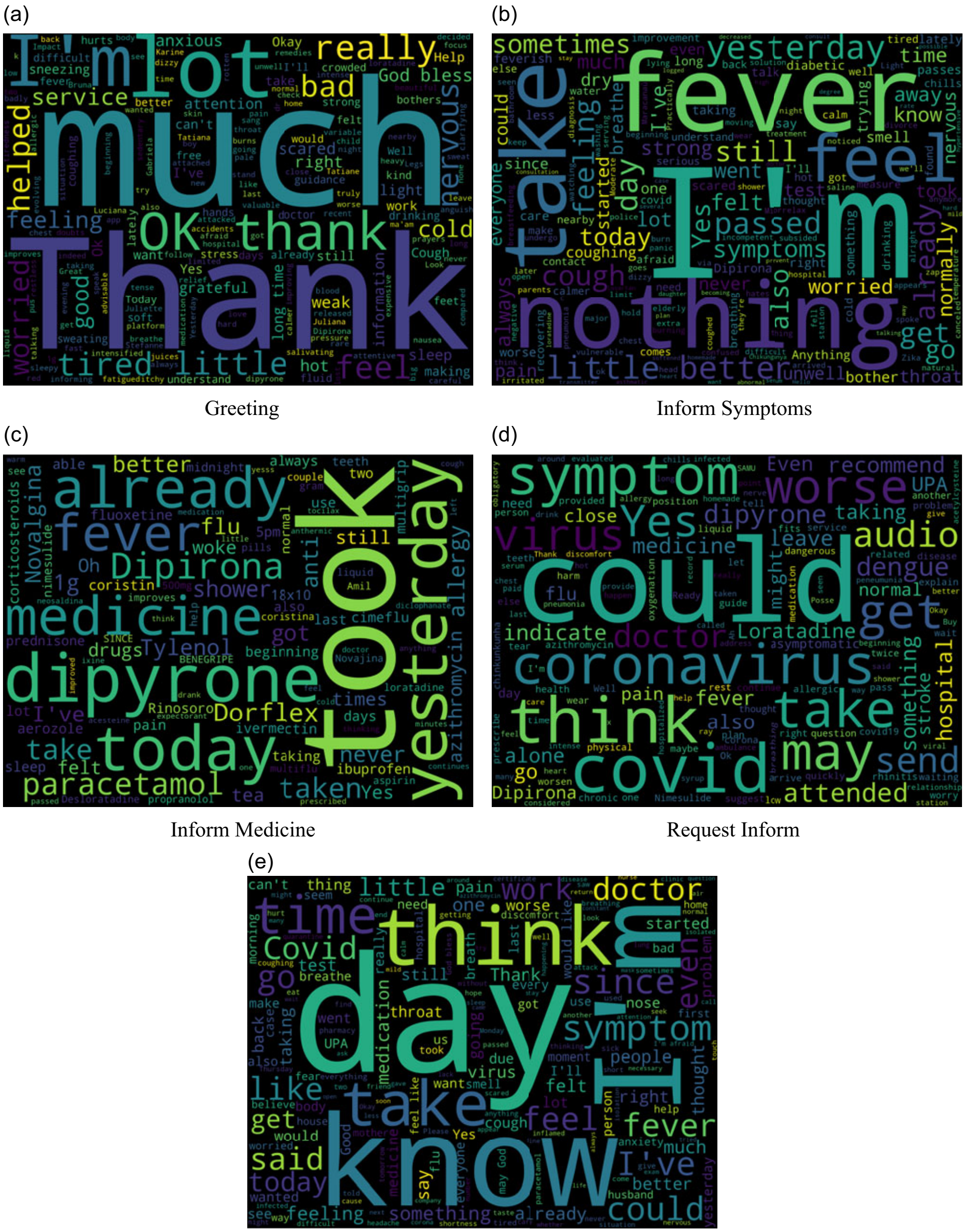


Figure 5. Silhouette score of each embedding model.

in which patients report symptoms. It is worth mentioning that the same intention can be found in more than one cluster. This is already expected since the value of the *SS* metric is close to 1, as shown in Table 3. By visual inspection, we decided to merge the clusters related to the same intention instead of having several clusters representing the same label.

Based on the inspections carried out through t-SNE, we assigned an intention to each cluster that best represented the intention present in the sentences of that cluster. We remind the reader that throughout the experimentation, only patient sentences are utilized, eliminating the necessity to differentiate the actor in the intention categories. Overall, the intentions identified in the clusters were as follows:

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press



Greeting

Inform Symptoms

Inform Medicine

Request Inform

Others

Figure 6. A word cloud generated by Glove embedding model representing sentences of one cluster intention.

1. **Greeting**: Sentences related to greetings or salutations, for example, when the patient says thank you, starts a conversation, or even says goodbye, in sentences such as *"Hello," "Good morning," and "Thank you very much."* Notice Figure 6a presents some words related to greetings.
2. **Inform Symptoms**: Phrases where patient symptoms are reported, for example, *"I dawned with a headache" and "A slight headache."* Notice Figure 6b presents the frequent symptoms and how the users described how they felt.

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

Table 4. Sentences of the Glove clusters representing an inform_symptoms intention

Sentence
I got tired when talking, and I totally lost my appetite, I don’t taste the food and I smell it,
but it’s not much.
I don’t know if other symptoms will appear. But for now, I just have body and back pain.
I sometimes feel shortness of breath and chest pain, but I never tried to find out what it
was I always took aerosol and it got better
Feeling sick, wanting to vomit. Here at home is me and my mother like this

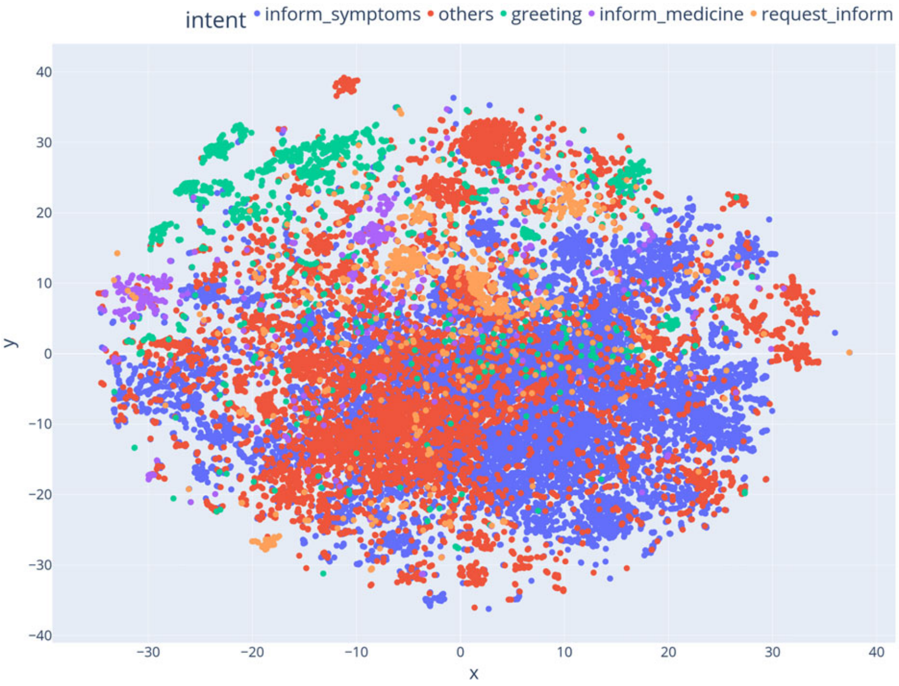


Figure 7. t-SNE visualization for the ninety-nine clusters generated with the Glove embedding model.

3. **Inform Medicine**: Phrases where the patient informs some medication they are taking, for example, *“I took paracetamol last night” and “I only take dipyrone.”* Observe Figure 6c shows the frequent words related to the medicines.
4. **Request Inform**: When the patient requests some information from the health professional, like in these examples: *“Is Dipyrone more effective?” and “Where do I get tested?”*. Figure 6d presents the frequent words used in the dialogs to request information about COVID-19.
5. **Others**: Cluster with difficulty identifying the primary intention or phrases that represent other types of intentions than the ones presented above. Figure 6e illustrates some frequent words from the cluster labeled as others.

Table 5 shows the distribution between the number of clusters labeled by type of intention found in each embedding model. Overall, our approach has shown that, even without

Table 5. Number of clusters by Intention

Intention	BERTimbau	FLAIR	Glove	LaBSE	MUSE
Greeting	14	14	14	7	4
Inform Symptoms	31	35	35	25	23
Inform Medicine	4	3	5	7	6
Request Inform	8	6	6	8	8
Others	34	37	39	34	41
Total	91	95	99	81	82

characterizing questions and answers throughout the conversation, we successfully labeled intentions in dialogs using unsupervised methods. This was achieved by representing sentences with an embedding model and subsequently clustering the resulting vectors. It's crucial to emphasize that this labeling method is automated. Nonetheless, in Section 6.4, we conduct an in-depth examination of the potential error introduced during the clustering phase.

6.2 Analysis of NLU model for intent classification

The experiments discussed in this section are related to the following research question: **(RQ2)** How to create an NLU model for intent classification using the semiautomatically labeled data from **(RQ1)**?

It is already expected that k-means may not identify outliers. Therefore, it is necessary to eliminate potential outliers within the clusters to improve the quality of the labeled data before using such data to train the NLU intent model. To accomplish this, we calculate the cosine distance between each sentence and the centroid of the cluster to which it belongs. Our outlier removal is grounded in this distance computation.

The sentences exhibiting a significant distance from the centroids of the clusters were excluded from the training dataset. This refinement, aimed at eliminating outliers and nonrepresentative sentences from the clusters, was implemented through the application of filters, namely *upper_bound_outiliers* and *upper_bound_median*, as described in Section 5.2.4. Notably, certain clusters were identified to contain sentences from diverse contexts, all labeled with the intent *others*. Consequently, alongside the initial refinement process, we filtered out clusters associated with the intention *others*, recognizing the potential noise introduced to the intent classifier. All in all, our evaluation of clustering quality goes through six scenarios detailed below: considering all sentences (**A**), incorporating various refinement strategies, and the presence (**B and C**) or absence (**D, E, and F**) of the intent *others*. The results of the DBS and SS are illustrated in Figures 8 and 9. The legend in these figures indicates the dataset used for calculation and evaluation purposes:

- **A**: Dataset with all sentences.
- **B**: Dataset refined by excluding sentences considered outliers, that is with a cosine distance to the centroid of their cluster exceeding the upper bound value (*upper_bound_outiliers*).
- **C**: Dataset refined by excluding sentences with a cosine distance to the centroid of their cluster exceeding the median value (*upper_bound_median*).
- **D**: Dataset refined by excluding the sentences associated with clusters labeled as *others* intention.

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

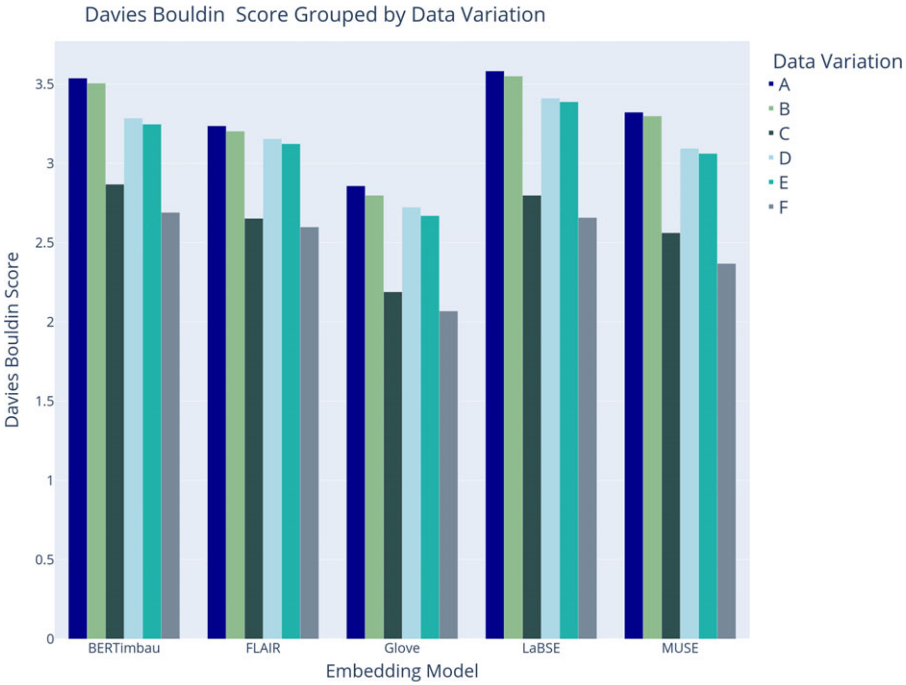


Figure 8. Davies Bouldin scores for datasets variations.

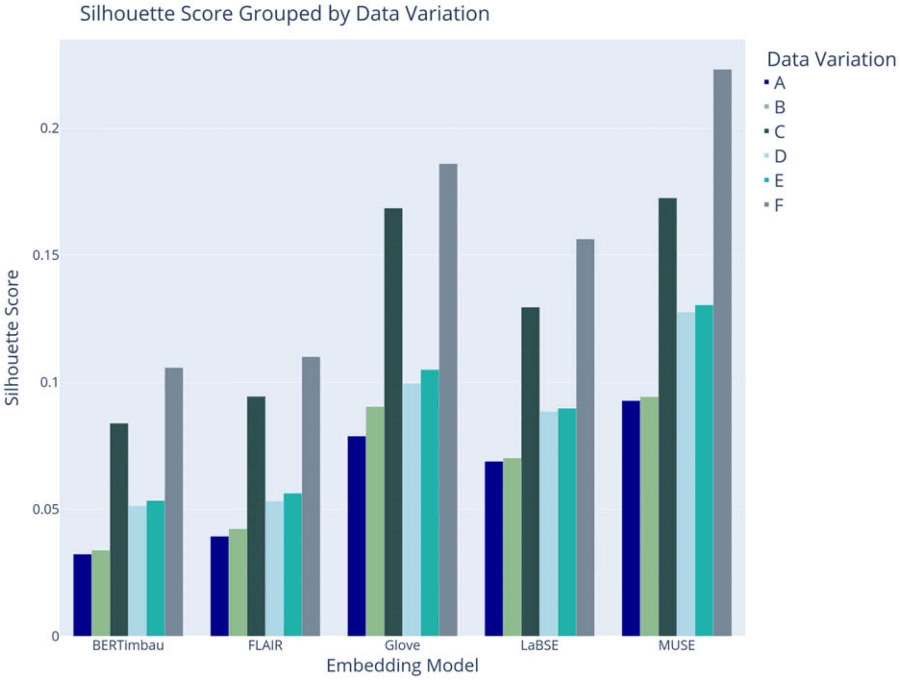


Figure 9. Silhouette scores for datasets variations.

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

Table 6. Number of sentences after outliers removal

Embedding	Number of sentences
BERTimbau	8837
FLAIR	8840
Glove	8900
LaBSE	8007
MUSE	6928

- **E**: Dataset subject to the same exclusion process as in **B and D**, where both outlier sentences and sentences belonging to clusters labeled as *others* are discarded.
- **F**: Dataset subject to the same exclusion process as in **C and D**, where both outlier sentences surpassing the median distance value and sentences from clusters labeled as *others* are eliminated.

By applying these filtering strategies, the data quality improved, as can be seen according to DBS and SS metrics. Figure 8 shows the variation of the DBS metric, and Figure 9 shows the variation of the SS metric for each embedding approach, whereas the data get closer to the centroid the DBS value improves (decreases the value, please compare the scenarios **A, B, and C**), as well as when the *others* intent is removed (please compare the scenarios **D, E, and F**), the values also improve. Similarly, for SS, when the values increase, there is an improvement in the quality of the clusters after the removal of the outliers.

We chose to construct the NLU intent model using dataset **F** as the training set, as it demonstrated the best values for DBS and SS. Remember, each embedding representation model produced other clusters and different outliers. So it is already expected that the number of sentences remaining within the clusters after removing the outliers would differ for each embedding model as shown in Table 6. In other words, the number of sentences in the **training/test** set varies for each embedding representation model. From the entire result set of sentences for each embedding model, as illustrated in Table 6, we conducted an analysis of the intersections between them. It was observed that only **1068** sentences were common among these models. Consequently, we randomly selected **300** sentences from this intersection for manual labeling. These labeled sentences constitute the validation set and will be excluded from both the **training** and **test** sets.

Table 7 shows the accuracy values achieved by the intent classification models built using the NN architecture presented in Section 5.2.5 implemented with Keras. The accuracy presented here is relative to the test data representing 30% of the dataset. All models achieved good performance, especially the ones generated with the representation of sentences through embeddings such as Glove, LaBSE, and MUSE, which also slightly outperform the other embeddings for DBS and SS. Based on our analysis, we found that the models we evaluated scored very well when we assessed the MCC metric. It's worth noting that a score of 1 on MCC represents a perfect prediction, so the models we evaluated came very close to achieving that ideal.

We also evaluate the classification models obtained from Rasa as explained in Section 5.2.5. Table 8 shows the values of Precision, Recall, F1-score, Accuracy, and MCC referring to the prediction of intentions over the test data. We notice that, again, the best intent classification models are generated using Glove, LaBSE, and MUSE as the embedding representation models. As we can see in Tables 7 and 8, on the classification metrics of intentions, the embedding representation used in the classification model that obtained the best values was MUSE, which is sentence embedding.

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

Table 7. Result Metrics (Macro) for the intent classification models based on feed-forward neural network

Embedding	Precision	Recall	F1-Score	Accuracy	MCC
BERTimbau	0.8912	0.9001	0.8954	0.9251	0.8677
FLAIR	0.9285	0.9150	0.9216	0.9450	0.8824
Glove	0.9307	0.9189	0.9244	0.9698	0.9116
LaBSE	0.9722	0.9676	0.9697	0.9749	0.9582
MUSE	**0.9846**	**0.9843**	**0.9844**	**0.9874**	**0.9792**

Table 8. Result Metrics (Macro) for the Rasa intent classification models

Embedding	Precision	Recall	F1-Score	Accuracy	MCC
BERTimbau	0.8842	0.8733	0.8785	0.9157	0.8454
FLAIR	0.9015	0.9014	0.9013	0.9372	0.8663
Glove	0.9163	0.9070	0.9116	0.9671	0.9028
LaBSE	0.9474	0.9370	0.9418	0.9546	0.9228
MUSE	**0.9575**	**0.9493**	**0.9530**	**0.9643**	**0.9408**

Figure 10 contains the histogram that shows the distribution of prediction of intentions for the model trained with MUSE. On the left side is the distribution of predictions correctly made, and on the right side, those made incorrectly. The predictions are distributed according to the confidence score. It can be seen that almost all predictions performed correctly had a confidence level equal to 100%. Still, a few incorrect predictions had the same confidence level, and this might be due to the significant overlap of clusters and distance of samples from different clusters shown during the application of the SS. However, this model has all the defined intentions and presents the best result compared to the other embeddings.

Based on the outcomes presented in this section, addressing the second research question (**RQ2**) involves exploring various outlier removal strategies and implementing two NNs to construct the intent classification model. These approaches produced promising results, with the NLU model achieving an MCC close to 1.

6.3 Analysis of the embedding representation

In this section, we would like to discuss the results guided by the research question: **(RQ3)** *Could the embedding representation of texts used for the clustering step and labeling assist the training of an intent classifier?* We aim to discover whether the embeddings employed to create the clusters can still effectively train the intent classifier, that is, using the embedding as a pretrained layer through the training network.

A research line in NLP offers comparative experimental results for the methods so that researchers can determine the most appropriate embedding for their problem based on the comparative analysis. The papers (Toshevska *et al.* 2020; Boggust *et al.* 2022) provide different comparisons between word embedding vectors to ensure the quality of word representation before use in an ML task. The evaluation methods are classified into two main categories: intrinsic and extrinsic (Zhai *et al.* 2016; Qiu *et al.* 2018). Intrinsic evaluation is independent of a specific NLP

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

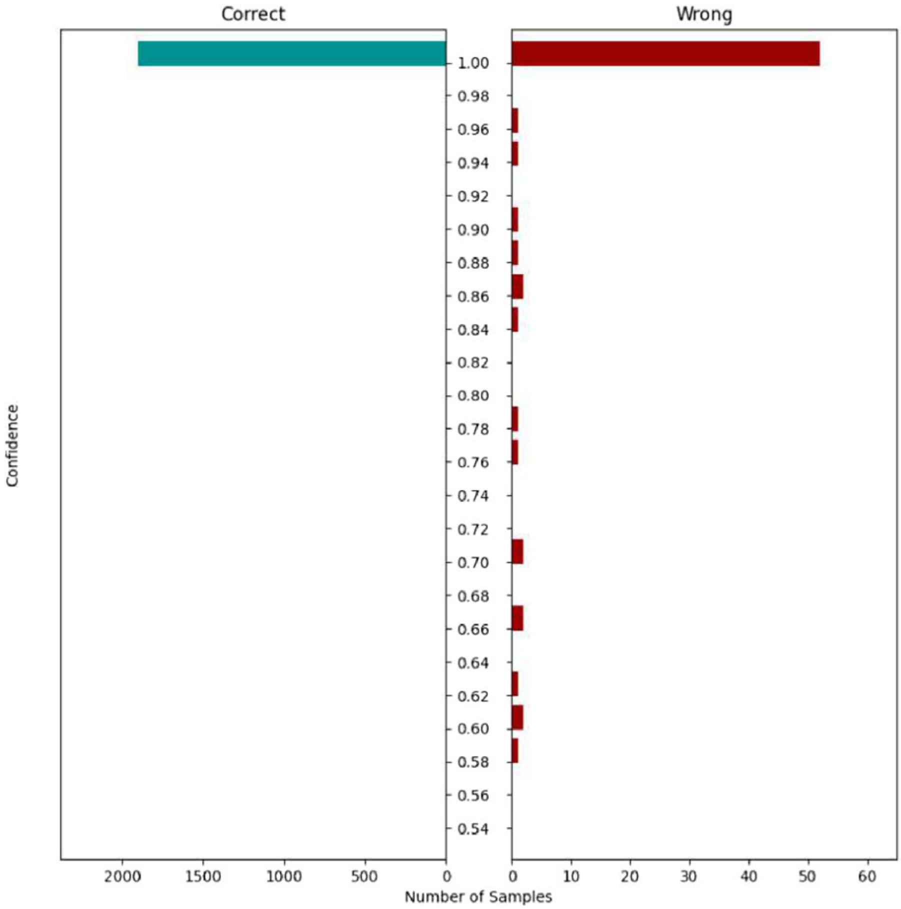


Figure 10. Histogram of prediction of intentions using the NLU trained with Rasa and MUSE embedding.

task; thus, it directly evaluates syntax or semantics relationships between words, for instance, evaluating the distance between words and sentences. The extrinsic method of word vectors is the evaluation integrated into an NLP task like natural language inference or sentiment analysis, chosen as an assessment method. Usually, word embedding evaluations collect accuracy and F1-score, among other metrics.

Considering the cluster quality, all the embedding representation models slightly perform the same (please, see Table 3); however, MUSE for DBS and SS metrics outperformed the others (please, see Figures 8 and 9) when we removed the outliers. The same applies to the accuracy of the intent classification models (Table 7 and 8) using MUSE to represent the sentences. So, the embedding models employed in these experiments demonstrate effective outcomes during the clustering phase. These findings align well because these embeddings continue to be effectively utilized as a pretrained embedding layer in the intent classification model network, especially MUSE.

6.4 Analysis of the potential labeling error introduced by clustering

In this section, we cover the experiments performed to analyze the research question: **(RQ4)** Given that clustering is an unsupervised technique, there is a possibility of introducing labeling errors during this step in the NLU intent classifier. Our intent classification model is trained

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

Table 9. Result Metrics (Macro) for the **validation set manually labeled** for the intent classification models based on feed-forward neural network

Embedding	Precision	Recall	F1-Score	Accuracy	MCC
BERTimbau	0.9427	0.8793	0.9073	0.9529	0.9158
FLAIR	0.9800	0.9164	0.9449	0.9630	0.9341
Glove	0.8769	0.7115	0.7564	0.8653	0.7575
LaBSE	0.9454	0.9360	0.9338	0.9630	0.9350
MUSE	0.9703	0.9376	0.9496	0.9630	0.9348

Table 10. Result Metrics (Macro) for the v**alidation set manually labeled** for the Rasa intent classification models

Embedding	Precision	Recall	F1-Score	Accuracy	MCC
BERTimbau	0.9325	0.7900	0.8347	0.9360	0.8854
FLAIR	0.9488	0.8908	0.9119	0.9360	0.8862
Glove	0.8796	0.6892	0.7352	0.8620	0.7521
LaBSE	0.9582	0.9419	0.9451	0.9663	0.9408
MUSE	0.9536	0.9182	0.9308	0.9596	0.9288

with data labeled using an unsupervised approach. It's crucial to note that in unsupervised learning, where the data used for learning lack information on the "correct" output, there is a risk of potential labeling errors being incorporated into the training set, which could mislead the intent classification model.

After removing outliers, we utilized **300** sentences from the intersection of datasets for each embedding model, as described in Section 6.2. This set also served as a validation set for the intent classification models. In this section, we employ the same set to analyze the potential errors introduced by the clustering phase in our approach. It's important to note that this validation set was manually annotated, and the text's intent labels do not come from the clustering phase. Tables 9 and 10 present the quality results achieved by the intent classification models when applied to the validation dataset for both the model based on the feed-forward NN and the one trained using Rasa.

As we can see, the model trained with the MUSE embedding obtained the best results. Similar to the results obtained with the labeled test data through clustering. However, it is important to highlight that the GLOVE decreased metric values using this validation dataset compared with the test data labeled through clustering.

Comparing Tables 7 and 9, and Tables 8 and 10, we can easily see that the clustering phase adds some labeling errors that misled the intent classifiers. To better understand labeling errors introduced in the clustering phase. Table 11 presents the distribution of the dataset manually labeled and used in the validation process. It is worth noting that this dataset was removed from the training and testing dataset. However, labels had been assigned to these sentences through the clustering process; these labels were not considered in the validation dataset, and the labels were added manually to the validation set. So Table 11 shows the number of sentences that had the cluster assigned correctly in the clustering phase compared to the total number of sentences used

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

Table 11. Number of correct label assignment by clustering of each embedding model in validation dataset

Intention	Ground Truth	BERTimbau	FLAIR	Glove	LaBSE	MUSE
Greeting	58	57	55	55	57	57
Inform Symptoms	182	176	181	179	182	182
Inform Medicine	14	14	12	8	14	14
Request Inform	43	34	32	14	32	32
Others	3	-	-	-	-	-
Total	300	281	280	256	285	285

for validation. Note that the embedding model that correctly assigned the lowest number of labels according to clustering was the Glove embedding model.

However, we still achieved competitive results in terms of accuracy for the trained models. Therefore, we can still profit from the dataset annotated by a clustering approach to train a COVID-19-based chatbot intent classifier. Our method is generic and can be applied to any other domain or disease, such as influenza and dengue.

It is worth mentioning that annotating a huge dataset is a challenging and time-consuming task. We have taken actions to mitigate the labeling error that could be introduced through the k-means clustering process. We have performed experiments with different implementations for the k-means algorithms and used different initializations. As well we have used word and sentence embeddings for comparison. We have used Silhouette analyses and the Davies-Bouldin index to adjust the number and size of clusters. We have manually inspected the clusters through word cloud visualization by different members of the research team. Additionally, we have removed outliers from our training set.

7. Discussion, limitations, and lessons learned

In this paper, we tackle the problem of how to label intentions in dialogs from a COVID-19-based chatbot and how to learn and create an NLU intention classification model for chatbots. We experimented with different embedding models to represent the texts over the dialogs to solve our problem. The embedding models analyzed were BERTimbau, FLAIR, Glove, LaBSE, and MUSE. The vector representations of the sentences were passed through a clustering algorithm (K-means) to group sentences with similar meanings into clusters. The labels were assigned through visual inspection (word clouds and t-SNE), referring to each cluster's intentions, thus performing semiautomatic labeling.

After labeling the data with the cluster (each one represents an intent), a data refinement process was applied to improve the quality of the datasets labeled by each embedding model. This refinement process consisted of discarding the sentences far (in terms of cosine distance) from the centroid of each cluster. Then, threshold-based approaches were applied. In one of them, only sentences classified as outliers were removed; in the other, we removed sentences with a distance to the centroid of their cluster greater than the median (considering the distance distribution within the cluster). The metrics referring to SS and DBS were evaluated in this process. At the end of this phase, it was proven that the cluster with only the sentences closest to the centroid obtained better results for these metrics.

After the refinement, intention classification models were built for each dataset labeled (according to the clusters). Besides, it is essential to note that each embedding model generates a different

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

cluster set. Each set of clusters is utilized for training an intent classification model. We experimented with two different neural architectures to train the intent classification models. One is based on the open-source framework Rasa, and another is based on deep NNs. After building the models, we also validated with data not seen by the models, that is, data manually labeled. It was noticed that the semiautomatic labeling (or clustering) included labeling error in the intention classification models. Because some of them obtained excellent results with the test data; however, with the manually labeled data, they got very different accuracy values. However, generally, the results obtained with all models built in the tests and the validation in terms of accuracy were above 86%. Furthermore, it is essential to point out that labeling a large amount of data for training these deep networks would be very time-consuming.

Reflecting on the challenges in adapting the proposed methodology to diverse domains and datasets within chatbot development, several lessons and guidelines have emerged. These insights provide valuable guidance that can be applied across domains and linguistic contexts. The major challenge may arise in acquiring domain-specific labeled datasets, as we faced in this work. So, first of all, practitioners should recognize the inherent variability in user intents and language across different domains, such as customer service, e-commerce, or healthcare. Domain-specific expertise is crucial for accurately annotating intentions in dialogs and ensuring that the clustering algorithm effectively captures the nuances of user interactions. Moreover, selecting or training embedding models tailored to the domain's vocabulary and linguistic characteristics is essential for generating meaningful representations of the text data. Additionally, practitioners must navigate ethical considerations regarding handling sensitive user data and ensuring compliance with privacy regulations, especially in domains like healthcare or finance.

Second, practitioners should carefully refine the labeled datasets to improve the data quality used for training intention classification models. The refinement process, which involves discarding sentences far from cluster centroids based on cosine distance, requires fine-tuning to account for the variability and noise inherent in different domains. Adjusting the threshold for discarding sentences and exploring alternative refinement techniques may be necessary to optimize model performance. Furthermore, practitioners must evaluate the effectiveness of intention classification models using metrics that capture the practical utility of chatbots in the specific domain, such as user satisfaction and task completion rate. Addressing these challenges and considerations will enable practitioners to effectively apply the approach in diverse chatbot contexts, resulting in the development of more accurate and domain-specific chatbot systems that cater to users' unique needs and preferences across various domains.

8. Conclusion and future work

This work shows an overview of a chatbot, its classifications, and how it can be used. We explain the chatbot's NLU component, its importance, and the building process. A dataset of dialogs among healthcare professionals and patients was used to apply unsupervised learning (clusterization) to find classes through visual inspection of the groups found. These classes were used to train NLU and intent classification models. The results of these models were evaluated, and in general, we achieved good results. Still, we used manually labeled data to validate the models, and there was a drop in the results, but they were still satisfactory.

In future works, we aim to investigate different strategies to label the dialogs, like applying topic modeling such as BERTopic (Grootendorst 2022). Each topic can represent an intention, for instance. We would also like to investigate other intention classification models COVID-19-based for chatbots and study whether transferring learning from such models to ours would be effective.

Acknowledgment. The research reported in this work was supported by FUNCAP project "Big Data Platform to Accelerate the Digital Transformation of Ceará State" under the number 04772551/2020.

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

References

Abdellatif A., Badran K., Costa D. and Shihab E. (2021). A comparison of natural language understanding platforms for chatbots in software engineering. In *IEEE Transactions On Software Engineering*, pp. 3087–3102.

Abdellatif A., Costa D., Badran K., Abdalkareem R. and Shihab E. (2020). Challenges in chatbot development: a study of stack overflow posts. In *Proceedings of the 17th international conference on mining software repositories*, pp. 174–185.

Adamopoulou E. and Moussiades L. (2020). An overview of chatbot technology. In *IFIP AIAI*. Springer, pp. 373–383.

Aggarwal C. C. and Reddy C. K. (2014). Data clustering. *Algorithms and applications. Chapman&Hall/CRC Data mining and Knowledge Discovery series, Londra.*

Aguiar L. A., Cruz L. A., da Silva T. L. C., do Carmo R. A. F. and Paixao M. H. E. (2022). Large-scale translation to enable response selection in low resource languages: a COVID-19 chatbot experiment. In *Anais Do XXXVII Simpósio Brasileiro De Bancos De Dados*. SBC, pp. 203–215.

Akbik A., Bergmann T., Blythe D., Rasul K., Schweter S. and Vollgraf R. (2019). Flair: an easy-to-use framework for state-of-the-art nlp. In *Proceedings of NAACL (Demonstrations)*, pp. 54–59.

Altman N. and Krzywinski M. (2018). The curse (s) of dimensionality. *Nature Methods* **15**, 399–400.

Boggust A., Carter B. and Satyanarayan A. (2022). *Embedding comparator: visualizing differences in global structure and local neighborhoods via small multiples*. In *27th International Conference on Intelligent User Interfaces*, pp. 746–766.

Cer D., Yang Y., Kong S.-y., Hua N., Limtiaco N., John R. S., Constant N., Guajardo-Céspedes M., Yuan S., Tar C. and Sung Y., Strope B. and Kurzweil R. (2018). Universal sentence encoder. *arXiv preprint arXiv:1803.1.1175*.

Conneau A., Kiela D., Schwenk H., Barrault L. and Bordes A. (2017). *Supervised learning of universal sentence representations from natural language inference data*. In *Proceedings of EMNLP*, pp. 670–680.

da Silva T. L. C., Lettich F., de Macêdo J. A. F., Zeitouni K. and Casanova M. A. (2020). Online clustering of trajectories in road networks. In *2020 21st IEEE International Conference on Mobile Data Management (MDM)*, IEEE, pp. 99–108.

Davies D. L. and Bouldin D. W. (1979). A cluster separation measure. *IEEE Transactions On Pattern Analysis and Machine Intelligence* **PAMI-1**, 224–227.

Defays D. (1977). An efficient algorithm for a complete link method. *The Computer Journal* **20**, 364–366.

Dos Santos Júnior V. O., Castelo Branco J. A., De Oliveira M. A., Coelho Da Silva T. L., Cruz L. A. and Magalhães R. P. (2021). A natural language understanding model covid-19 based for chatbots. In *2021 IEEE 21st International Conference on Bioinformatics and Bioengineering (BIBE)*, pp. 1–7.

Ebadi A., Xi P., Tremblay S., Spencer B., Pall R. and Wong A. (2021). Understanding the temporal evolution of covid-19 research through machine learning and natural language processing. *Scientometrics* **126**, 725–739.

Ester M., Kriegel H.-P., J. S. and Xu X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, **96**, pp. 226–231.

Fazzinga B., Galassi A. and Torroni P. (2021). An argumentative dialogue system for covid-19 vaccine information.

Feng F., Yang Y., Cer D., Arivazhagan N. and Wang W. (2020). Language-agnostic bert sentence embedding. *arXiv preprint arXiv:2007.01852*.

Gaglo K., Degboe B. M., Kossingou G. M. and Ouya S. (2021). Proposal of conversational chatbots for educational remediation in the context of covid-19. In *2021 23rd International Conference on Advanced Communication Technology (ICACT)*, pp. 354–358.

Galassi A., Lippi M. and Torroni P. (2021). Attention in natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, **32**(10), 4291–4308.

Gao J., Galley M. and Li L. (2018). Neural approaches to conversational ai. In *The 41st International ACM SIGIR*, pp. 1371–1374.

Grootendorst M. (2022). Bertopic: Neural topic modeling with a class-based tf-idf procedure. arXiv preprint arXiv:2203.05794.

Ham J. and Kim E.-S. (2021). Semantic alignment with calibrated similarity for multilingual sentence embedding. In Moens M.-F., Huang X., Specia L. and Yih S. W.-t. (eds), *Findings of the Association for Computational Linguistics: EMNLP*. Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 1781–1791.

Han B., Liu L. and Omiecinski E. (2012). Neat: Road network aware trajectory clustering. In *2012 IEEE 32nd International Conference on Distributed Computing Systems*, IEEE, pp. 142–151.

Harris Z. S. (1954). Distributional structure. *Word-Journal of the International Linguistic Association* **10**, 146–162.

Hien H. T., Cuong P.-N., Nam L. N. H., Nhung H. L. T. K. and Thang L. D. (2018). Intelligent assistants in higher-education environments: the fit-ebot, a chatbot for administrative and learning support. In *Proceedings of the ninth international symposium on information and communication technology*, pp. 69–76.

Hoaglin D. C. and Iglewicz B. (1987). Fine-tuning some resistant rules for outlier labeling. *Journal of the American Statistical Association* **82**, 1147–1149.

Iyyer M., Manjunatha V., Boyd-Graber J. and Daumé III, H (2015). Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd ACL-IJCNLP (volume 1: Long papers)*, pp. 1681–1691.

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

Judson T. J., Odisho A. Y., Young J. J., Bigazzi O., Steuer D., Gonzales R. and Neinstein A. B. (2020). Implementation of a digital chatbot to screen health system employees during the COVID-19 pandemic. *Journal of the American Medical Informatics Association* **27**, 1450–1455.

Kassambara A. (2017). Practical guide to cluster analysis in R. In *Unsupervised Machine Learning*, **1**. Sthda

Khanna A., Pandey B., Vashishta K., Kalia K., Pradeepkumar B. and Das T. (2015). A study of today's ai through chatbots and rediscovery of machine intelligence. *International Journal of u-and e-Service, Science and Technology* **8**, 277–284.

Klein A. Z., Magge A., O'Connor K., Amaro J. I. F., Weissenbacher D. and Hernandez G. G. (2021). Toward using twitter for tracking covid-19: a natural language processing pipeline and exploratory data set. *Journal of Medical Internet Research* **23**, e25314.

Kucherbaev P., Bozzon A. and Houben G.-J. (2018). Human-aided bots. *IEEE Internet Computing* **22**, 36–43.

Kushwaha A. K., Kumar P. and Kar A. K. (2021). What impacts customer experience for b2b enterprises on using ai-enabled chatbots? insights from big data analytics. *Industrial Marketing Management* **98**, 207–221.

Le Q. and Mikolov T. (2014). Distributed representations of sentences and documents. In *ICML*. PMLR, pp. 1188–1196.

Lee J. Y. and Dernoncourt F. (2016). Sequential short-text classification with recurrent and convolutional neural networks.

Lei H., Lu W., Ji A., Bertram E., Gao P., Jiang X. and Barman A. (2021). Covid-19 smart chatbot prototype for patient monitoring.

Li I., Li Y., Li T., Alvarez-Napagao S., Garcia-Gasulla D. and Suzumura T. (2020). What are we depressed about when we talk about covid-19: Mental health analysis on tweets using natural language processing. In *International Conference on Innovative Techniques and Applications of Artificial Intelligence*, Springer, pp. 358–370.

Liu J., Li Y. and Lin M. (2019a). Review of intent detection methods in the human-machine dialogue system. In *Journal of Physics: Conference Series*, **1267**, IOP Publishing, pp. 012059.

Liu W., Wang Z., Liu X., Zeng N., Liu Y. and Alsaadi F. E. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing* **234**, 11–26.

Liu X., He P., Chen W. and Gao J. (2019b). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, pp. 4487–4496.

Luo B., Lau R. Y., Li C. and Si Y.-W. (2022). A critical review of state-of-the-art chatbot designs and applications. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* **12**, e1434.

Matthews B. (1975). Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta (BBA) - Protein Structure* **405**, 442–451.

Mikolov T., Sutskever I., Chen K., Corrado G. S. and Dean J. (2013). Distributed representations of words and phrases and their compositionality. In *NIPS*, pp. 3111–3119.

Miner A. S., Laranjo L. and Kocaballi A. B. (2020). Chatbots in the fight against the covid-19 pandemic. *NPJ Digital Medicine* **3**, 1–4.

Naseem U., Razzak I., Khan S. K. and Prasad M. (2021). A comprehensive survey on word representation models: from classical to state-of-the-art word representation language models. *Transactions On Asian and Low-Resource Language Information Processing* **20**, 1–35.

Nassif A. B., Shahin I., Attili I., Azzeh M. and Shaalan K. (2019). Speech recognition using deep neural networks: a systematic review. *IEEE Access* **7**, 19143–19165.

Nimavat K. and Champaneria T. (2017). Chatbots: An overview. types, architecture, tools and future possibilities. *International Journal for Scientific Research & Development* **5** 1019–1024.

Ouerhani N., Maalel A., Ghézala H. B. and Chouri S. (2020). Smart ubiquitous chatbot for covid-19 assistance with deep learning sentiment analysis model during and after quarantine.

Peikari M., Salama S., Nofech-Mozes S. and Martel A. L. (2018). A cluster-then-label semi-supervised learning approach for pathology image classification. *Scientific Reports* **8**, 1–13.

Pennington J., Socher R. and Manning C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of EMNLP*, pp. 1532–1543.

Qiu Y., Li H., Li S., Jiang Y., Hu R. and Yang L. (2018). Revisiting correlations between intrinsic and extrinsic evaluations of word embeddings. In *Chinese Computational Linguistics and Natural Language Processing Based On Naturally Annotated Big Data*. Springer, pp. 209–221.

Rasa (2022). Introduction to rasa open source & rasa pro. Available at: https://rasa.com/docs/rasa (Accessed 10 October 2022).

Reimers N. and Gurevych I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.

Rousseeuw P. J. (1987). *Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics* **20**, 53–65.

Souza F., Nogueira R. and Lotufo R. (2020). BERTimbau: pretrained BERT models for Brazilian Portuguese. In 9th BRACIS.

Toshevska M., Stojanovska F. and Kalajdjieski J. (2020). Comparative analysis of word embeddings for capturing word similarities. arXiv preprint arXiv:2005.03812.

Van der Maaten L. and Hinton G. (2008). Visualizing data using t-sne. *Journal of machine learning research* **9**.

https://doi.org/10.1017/nlp.2024.45 Published online by Cambridge University Press

Vassilvitskii S. and Arthur D. (2006). *k-means++: The advantages of careful seeding.* In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1027–1035.

Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser Ł. and Polosukhin I. (2017). Attention is all you need. In NIPS, pp. 5998–6008.

Wang L. L., Lo K., Chandrasekhar Y., Reas R., Yang J., Eide D., Funk K., Kinney R., Liu Z., Merrill W., Mooney P., Murdick D., Rishi D., Sheehan J., Shen Z., Stilson B., Wade A., Wang k, Wang N. X. R., Wilhelm C., Xie B., Raymond D., Weld D. S., Etzioni O. and Kohlmeier S. (2020). Cord-19: The covid-19 open research dataset. *ArXiv*.

Wang Y.-Y., Deng L. and Acero A. (2005). Spoken language understanding. *IEEE Signal Processing Magazine* **22**, 16–31.

Weld H., Huang X., Long S., Poon J. and Han S. C. (2021). A survey of joint intent detection and slot filling models in natural language understanding. *ACM Computing Surveys (CSUR)* **55**.

Wieting J., Bansal M., Gimpel K. and Livescu K. (2016). Towards universal paraphrastic sentence embeddings.

Yang Y., Cer D., Ahmad A., Guo M., Law J., Constant N., Abrego G. H., Yuan S., Tar C., Sung Y.-H.,Strope B. and Kurzweil R. (2019). Multilingual universal sentence encoder for semantic retrieval. arXiv preprint arXiv:1907.

Yao K., Zweig G., Hwang M.-Y., Shi Y. and Yu D. (2013). Recurrent neural networks for language understanding. In Interspeech, pp. 2524–2528.

Zhai M., Tan J. and Choi J. (2016). Intrinsic and extrinsic evaluations of word embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 4282–4283.

Cite this article: dos Santos Júnior V O, de Oliveira M A, Cruz L A and Coelho da Silva Ticiana L.. Learning and semi-automatic intention labeling for classification models: a COVID-19 dialog attendance study for chatbots. *Natural Language Processing* https://doi.org/10.1017/nlp.2024.45