Intelligent Systems with Applications 16 (2022) 200113

Contents lists available at ScienceDirect

Intelligent Systems with Applications

journal homepage: www.journals.elsevier.com/intelligent-systems-with-applications

# A privacy-preserving dialogue system based on argumentation

Bettina Fazzinga [a], Andrea Galassi [*,b], Paolo Torroni [b]

[a] DICES, University of Calabria, Via Pietro Bucci, Rende, 87036, Italy
[b] DISI, University of Bologna, Viale Risorgimento 2, Bologna, 40136, Italy

ARTICLE INFO

*Keywords:*
Dialogue systems
Argumentation
Data protection
Explainable AI
Trustworthy AI

*MSC:*
68T50
68T35
68T99

*PACS:*
0705

ABSTRACT

Dialogue systems are a class of increasingly popular AI-based solutions to support timely and interactive communication with users in many domains. Due to the apparent possibility of users disclosing their sensitive data when interacting with such systems, ensuring that the systems follow the relevant laws, regulations, and ethical principles should be of primary concern. In this context, we discuss the main open points regarding these aspects and propose an approach grounded on a computational argumentation framework. Our approach ensures that user data are managed according to data minimization, purpose limitation, and integrity. Moreover, it is endowed with the capability of providing motivations for the system responses to offer transparency and explainability. We illustrate the architecture using as a case study a COVID-19 vaccine information system, discuss its theoretical properties, and evaluate it empirically.

## 1. Introduction

Dialogue systems are among the most popular forms of commercial AI products. In the 2019 Gartner CIO Survey, CIOs identified chatbots as the main AI-based application used in their enterprises,[1] with a global market valued in the billions of USD.[2]

In fact, chatbots are one example of the extent AI technologies are becoming ever more pervasive, both in addressing global challenges, and in the day-to-day routine. Public administrations too are adopting chatbots for key actions such as helping citizens in requesting services[3] and providing updates and information, for example, in relation with COVID-19 (Amiri and Karahanna, 2022; Miner et al., 2020).

However, the expansion of intelligent technologies has been met by growing concerns about possible misuses, motivating a need to develop AI systems that are trustworthy. On the one hand, governments are pressured for gaining or preserving an edge in intelligent technologies, which make intensive use of large amounts of data. On the other hand, there is an increasing awareness of the fundamental need for data protection regulations. To make matters more complicated, different jurisdictions have different data protection regulations. Indeed, threats to the privacy of individuals are real. For example, a recent uproar was caused by Singapore's admission that data from its COVID-19 contact tracing programme could also be accessed by police, reversing earlier privacy assurances.[4] All this motivates the need for a trustworthy AI.

According to several studies, including the Ethics guidelines produced by the High-Level Expert Group on AI,[5] trustworthy AI systems need not only be robust, but also respectful of all applicable laws and regulations, as well as of ethical principles and values. Among the tenets of trustworthy AI are *privacy and data governance, transparency*, and *auditability*. For dialogue systems in particular, a study by Saglam et al. (2021) shows that a major reason of distrust in them is the user's loss of *agency* over the data they provide to the system.

Concerning privacy and data governance, in the context of chatbots we believe that a legitimised access to data should be ensured by an architectural design that takes data access into account from the very beginning, preserving the user's agency over their data. This is

* Corresponding author.
*E-mail address:* a.galassi@unibo.it (A. Galassi).

[1] https://www.gartner.com/smarterwithgartner/chatbots-will-appeal-to-modern-workers/.
[2] https://www.mordorintelligence.com/industry-reports/chatbot-market.
[3] https://www.canada.ca/en/employment-social-development/services/my-account/terms-use-chatbot.html.
[4] https://www.bbc.com/news/world-asia-55541001.
[5] https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai.

https://doi.org/10.1016/j.iswa.2022.200113
Received 21 April 2022; Received in revised form 22 July 2022; Accepted 12 August 2022
Available online 18 August 2022

especially true of applications that necessitate the interaction among different legal entities. For example, consider a government's chatbot, giving citizens information about COVID-19 vaccines. Let us say that such a chatbot relies on a transnational, or regional agency that contributes medical expertise on the subject. To provide relevant information, the system needs eliciting user personal, possibly sensitive data. It is thus extremely important that data processing is limited to the specific purpose that matches the user's need, and that only the necessary user data is stored and transmitted. Moreover, the user should be able to investigate and understand the reasons behind the AI system's recommendations, without having to be technically savvy. Finally, we think that auditability is especially important when an AI chatbot has to deal with personal data and offer advice: in particular, the models and algorithms used to produce such an advice should be transparent and verifiable.

We thus propose a dialogue system architecture inspired by the principles and values of trustworthy AI, that explicitly addresses the above points in the following way:

- user interaction is carried out *in natural language*, not only for providing information to the user, but also to answer user queries about the *reasons* leading to the system output (explainability);
- the system selects answers based on a *transparent reasoning module*, built on top of a computational argumentation framework with a *rigorous, verifiable semantics* (transparency, auditability);
- the treatment of user data is made in accordance with the data minimization, purpose limitation and storage limitation principles. To this end, the natural language interface and the reasoning module are *decoupled* so as to ensure that no personal data is passed from one module to the other (privacy and data governance).

The present paper demonstrates the feasibility of the approach and shows its workings via an illustration consisting of an AI chatbot aimed to give advice on COVID-19 vaccines. Our goal here is to move one step towards bridging the gap between fundamental, perhaps abstract, ethical principles, and practical AI applications.

In our previous works (Fazzinga et al., 2021a; 2021b), we presented a very general and preliminary idea of our framework with no formalization, and provided a very preliminary evaluation. Here we present a detailed and complete overview of our architecture, describing each part in detail, present the algorithm and the formal properties of our strategy, and include a more broad and robust evaluation.

The paper is structured as follows: in Section 2 we present the legal background and the related works. In Section 3 we propose our approach, while in Section 4 we discuss an illustration on COVID-19 vaccines. Section 5 evaluates the approach and Section 6 concludes.

## 2. Background

Our proposal of a privacy-preserving AI dialogue system builds on recent advances in language technologies (dialogue systems), knowledge representation (argumentation as a framework for non-monotonic reasoning and explainability), and on the latest European regulations in terms of data protection. In this section, we provide the necessary background and comment on the most significant differences of our proposal from relevant work.

### 2.1. Data protection in the E.U.

In the European Union, every processing of personal data, in any context, is subject to a set of rules and principles that impose obligations on those who process data and attribute rights to those the personal data is referred to. The E.U.'s General Data Protection Regulation[6] (GDPR) has general applicability and not only defends the fundamental right to data protection, but also aims to protect all the fundamental rights and freedoms that are implicated by the processing of personal data (Hildebrandt, 2020).

Especially relevant for the purposes of this article is GDPR's Article 5. In particular, Article 5(1) states the principles that must guide the processing of personal data. These include *purpose limitation, data minimization, storage limitation*, and *integrity and confidentiality*. The former three impose that the collection and process of personal data must be limited only to what is strictly necessary to fulfill a specified purpose and must not be kept or processed further, while the latter aims to guarantee the appropriate security of personal data. Article 5(2) instead covers accountability and specifies that the *data controller*, the legal body which determines the purposes and means of the processing of personal data, must be able to demonstrate compliance with all these principles.

### 2.2. Document redaction and sanitization

The anonymization of unstructured textual data is still an open and challenging task (Batet and Sánchez, 2018; Lison et al., 2021). Various approaches have been proposed. *Redaction* is the processing of textual document with the aim to remove personal sensitive information (Szarvas et al., 2007). *Sanitization* instead replaces such information with more general and impersonal variations (Chakaravarthy et al., 2008). Unfortunately, current redaction and sanitization technology is still far from guaranteeing zero-risk to the user (Li et al., 2017). Common solutions only focus on predefined categories of entities. If they can certainly serve as useful privacy-enhancing techniques, they do not qualify as full anonymization as defined by regulations like the GDPR, because they ignore elements that may play a role in re-identifying the individual (Lison et al., 2021; Pilán et al., 2022). Moreover, the most successful approaches are limited to specific domains, and often rely on large, hard-to-obtain datasets (Hassan et al., 2019; Iwendi et al., 2020; Nguyen and Cavallari, 2020; Sánchez et al., 2014).

Our approach distinguishes itself from the above, because it does not share or save user information, but it replaces it by a set of general, "sanitized" information elements that are pertinent to the use case.

### 2.3. Dialogue systems

Although terminology varies widely, there is a generally accepted distinction between conversation-oriented and task-oriented dialogue systems. Conversational agents aim to support open-domain dialogues. They are commonly called *chatbots*. Task-oriented dialogue systems instead aim to assist the user in completing well-defined tasks in a given domain (Chen et al., 2017; Deriu et al., 2021). Such tasks may consist in performing a specific action or eliciting user information, or, like in our case, providing information to the user.

The dialogue response typically involves four stages: understanding the question, managing the dialogue, optionally performing an action, and generating the answer. While these steps can be managed separately with pipeline architectures, much of the recent literature regards end-to-end neural approaches. Conventional pipeline architectures have problems propagating the user's feedback across all the components, while the advantage of modularity is thwarted by the interdependence of such modules, hampering the adaptation of a pipeline system to a new domain or a new task (Wen et al., 2017; Zhao and Eskénazi, 2016). The success of deep learning architectures at many NLP task, among which natural language understanding and generation (Galassi et al., 2021; Young et al., 2018), has motivated researchers to extensively research their application in this area (Luo et al., 2019; Mohamad Suhaili et al., 2021; Rajendran et al., 2018). The downside of such approaches is that they are usually costly to implement for a new task because they require large training datasets. Moreover, data-driven dialogue systems are subject to bias (Barikeri et al., 2021; Dinan et al., 2020; Liu et al., 2020), privacy violations, adversarial examples, and several other ethical issues

[6] https://eur-lex.europa.eu/eli/reg/2016/679/oj.

and safety concerns. Henderson et al. (2018) provide a broad and thorough discussion.

Given our focus on user protection and our aim to develop a general, data-independent approach, we have opted for an architecture that does not involve training, and is modular. Our approach uses techniques common in Information Retrieval-based dialogue systems, where the user's sentences are treated as queries and answers are retrieved from a knowledge base made of dialogues. For example, Charras et al. (2016) compare the use of cosine similarity between TF-IDF representations of sentences, and specifically trained *doc-to-vec* embeddings (Le and Mikolov, 2014). Likewise, we measure the similarity between sentence embeddings to match the user sentences with the ones provided by the knowledge base. However, we rely on pre-trained models and on sanitized text produced by domain experts.

To the best of our knowledge, we are the first ones to introduce a dialogue system architecture whose design goal is to guarantee user data protection. In fact, previous work regarding sensitive, health data related (Brixey et al., 2017; Xu et al., 2019) do not specifically address user and data protection.

Regarding the use of argumentation in the context of dialogue systems, previous works mainly focused on persuasion. Rosenfeld and Kraus (2016) rely on reinforcement learning, while Rach et al. (2018) envision the dialogue as a game and the answers as moves along a previously defined scheme. In both cases, the agents are limited in their input and outputs to the sentences present in the knowledge base. Chalaguine and Hunter (2020) propose a persuasive dialogue system to convince students of the importance of university fees. The approach is based on a knowledge graph where each possible user sentence is encoded in a node and all its answers are linked to it by an edge. When the user writes a sentence, the system searches for an argument in the graph that matches with the user sentence, and chooses an answer to give to the user among the nodes linked to it. Although this system is surely related to the one we propose, it also differs significantly in that the choice of the answer is made "locallyg, by only looking at the possible answers to the last question posed by the user, while ignoring what was said earlier. The lack of a history of the conversation, with the ensuing impossibility of retrieving multiple pieces of information within a single exchange, strongly limit the application of such a system in real-world scenarios where dialogues naturally take into account things said at different times. Another limitation of these approaches is their relying on *lexical*, instead of *semantic* similarity.

### 2.4. Dialogue systems for COVID-19 pandemic

Chalaguine and Hunter (2021) developed a persuasive chatbot to persuade users to get vaccinated against Covid-19, using the same techniques employed in their previous work. Dos Santos Júnior et al. (2021) focused on natural language understanding and study the use of embeddings and clustering algorithms to automatically annotate a datasets of covid-related conversations with intentions labels. The VIRA system (Gretz et al., 2022) is trained to recognize the intent of the user among a set of candidates, and replies with one of the corresponding answers. Altay et al. (2021) studied the positive impact of the use of chatbots, Judson et al. (2020) highlighted the users' suspicions, while Schubel et al. (2021) reported differences in their use by different population subgroups.

None of these works addresses handling user sensitive data and related privacy issues. Moreover, besides some focus on persuasive dialogue systems, to best of our knowledge no work has been done on systems for providing reliable information to the users.

More detailed information regarding the use of chatbots in the Covid-19 public health response can be found in the survey by Amiri and Karahanna (2022).

### 2.5. Argumentation frameworks

Abstract Argumentation (AA) (Dung, 1995) is a branch of Artificial Intelligence (AI) that gained significant attention in the last years due to its capability of modelling debates, dialogues, and, in general, situations where conflicts and diversity of opinions arise. One important point that leads to the usage of AA as a reasoning mechanism at the core of several dialogue-based applications in AI is also its natural aptitude to provide æexplanationsg (Modgil et al., 2013). Indeed, in recent years, the capability of providing motivations for systems/agents behaviours has become crucial in AI, and AA is taking on an increasingly central role (Chesñevar et al., 2020; Cyras et al., 2021). In fact, modelling a dispute/dialogue as an AA framework not only offers the possibility of locating the arguments that represent a good/bad point in a rebuttal, but has the further advantage of possibly providing a æwitnessg of the reason why a certain argument is a good/bad point. From a technical standpoint, the disputes in AA are modelled as graphs, where the arguments, that are the sentences claimed by the agents participating the dispute, are the nodes, and the conflicts/contradictions between the sentences, named attacks, are the edges of the graph. As an example, consider the following scenario. Andrea says argument $a$: æMilan is a very livable cityg. Matt says argument $b$: æMilan is one of the most polluted cities of the world, so it is absolutely not livableg. Alice says argument $c$: æSeveral parameters are used to establish whether a city is livable, thus you cant say that Milan is not livableg. This scenario can be modelled as the AA graph $\langle A,D\rangle$, where $A$ consists of the arguments $a, b, c$ and $D$ consists of the edges $(a,b),(b,a),(c,b)$.

A lot of work has been devoted to reasoning over argumentation graphs (Baroni and Giacomin, 2009; Charwat et al., 2015; Fazzinga et al., 2019), and several ways of identifying ærobustg arguments or sets of arguments have been proposed, called semantics (Dung, 1995; Dung et al., 2007). A popular one is the *admissible* semantics. It stipulates that a set $S$ of arguments is an admissible extension (that is, it conforms to the admissible semantics) if and only if (i) $S$ is conflict-free, i.e. there is no attack between arguments in $S$ and (ii) $S$ *defends* every argument in it, i.e., $S$ attacks every argument (outside $S$) attacking arguments in $S$. Condition (ii) reveals that the admissible semantics is based on the fundamental concept of acceptance: to be an admissible extension, every argument $a$ of $S$ must be *acceptable* w.r.t. it, meaning that $S$ must counterattack every attack from outside towards $a$. Continuing the above example, both $S_1 = \{c\}$ and $S_2 = \{a, c\}$ are admissible extensions, while $S_3 = \{a, b\}$ and $S_4 = \{b, c\}$ are not, as they are not conflict-free, and neither is $S_5 = \{b\}$ as $b$ is not acceptable w.r.t. $S_5$: in fact $S_5$ does not defend $b$ against the attack from $c$.

## 3. Architecture and methods

To illustrate our proposal, let us consider a government intending to provide a personalized information service to its citizens through a dialogue system. The interaction between the user and the system would be similar in many different scenarios, while the back-end retrieval of information would depend on the specific case. It is also reasonable to imagine scenarios where the knowledge base used by the service to retrieve the specific information may not be maintained or owned by the entity itself, but instead by a third party, such as a transnational agency contributing specialized medical expertise. In such cases, where the interaction with the knowledge base is handled by a third party, the provider of the dialogue system must guarantee to the users the protection of their personal information. The third party's access to them must be limited, both in terms of content and time, to what is strictly necessary for computing the pertinent answer. It is, therefore, the responsibility of the service provider to ask the user for the relevant information, to analyze and process them, to guarantee that any information that is irrelevant (but potentially sensitive) is removed, and to guarantee that the set of maintained data as a whole guarantees anonymity.

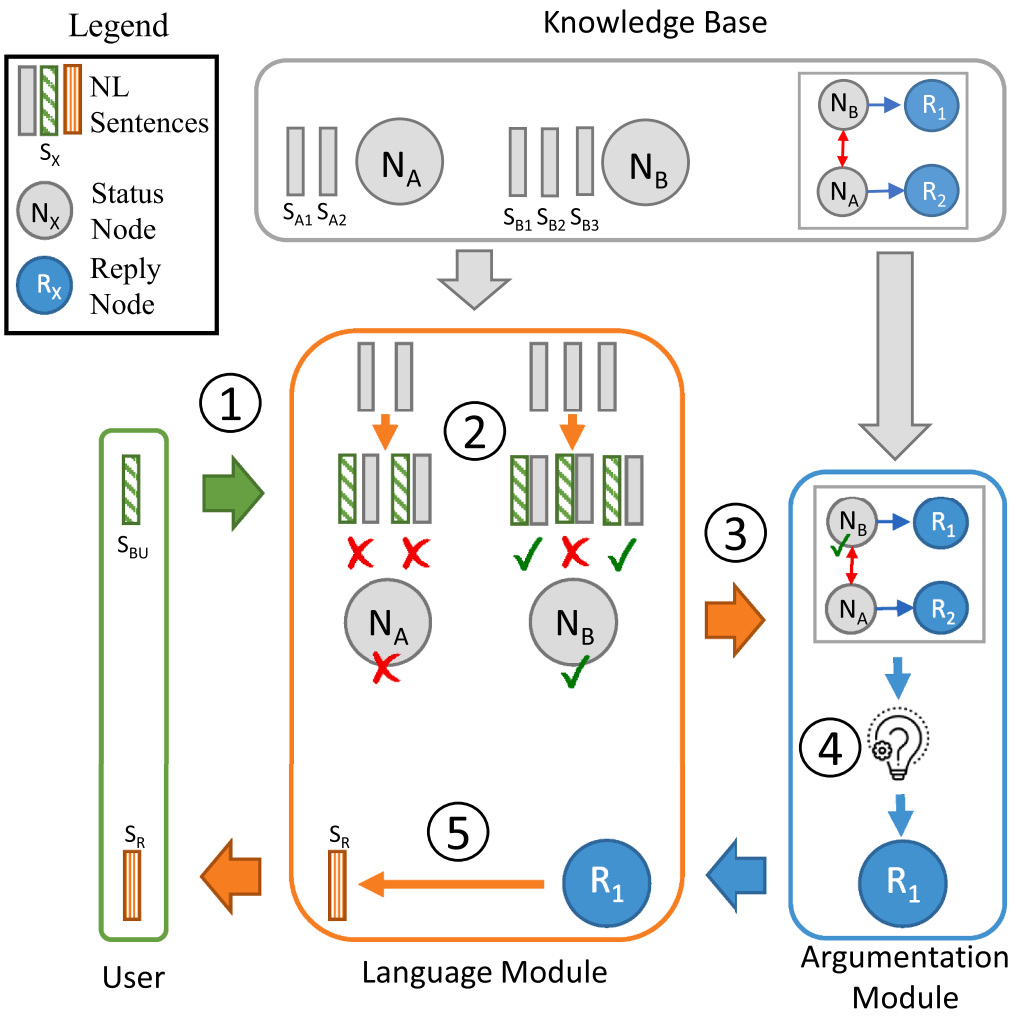


**Fig. 1.** System architecture and example of interaction with the user. Sentences are represented as rectangles and indicated with *S*, while circles are used for status and reply nodes (indicated respectively with *N* and *R*). We represent a case where nodes and sentences refers to two concepts, *A* and *B*, and the user sentence regards *B*. The information provided by the user is represented with the green color and by diagonal stripes. It is easy to see that such information does not reach the argumentation module. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Consequently, we propose a modular architecture for dialogue systems made by the following components:

- a **Knowledge Base** (KB) made by experts, containing all the possible relevant cases, answers, and relationships between them;
- a **Language module** that processes the user's input, including sensitive information, and maps it to the corresponding KB cases;
- an **Argumentation module** for reasoning over such KB cases and computing answers.

The dialogue process is shown in Fig. 1 and can be summarized as follows:

1. The user interacts with the Language module, providing personal information needed to satisfy their request.
2. The language module compares the user information with the KB to understand which cases are relevant.
3. The language module establishes a connection with the Argumentation module and provides it an anonymous, sanitized, and generalized version of the user's information.
4. The Argumentation module elaborates the information and computes an answer.
5. The Argumentation module sends the answer to the Language module, which forwards it to the user, optionally processing it further. Such an answer may be the information required by the user or a request for further personal information that is needed to provide a proper answer.
6. In case more information is required the process goes back to point 2.
7. In case the user has received the final answer, they can ask for a detailed explanation of the answer, which will be provided by the Argumentation module.
8. As soon as the user decides to terminate the interaction, the connection between the modules is closed, and all the information related to the user is deleted.

It is important to highlight that the exchange of personal and sensitive data occurs only between the user and the Language Module. The Argumentation module has access only to a general and broad representation that is strictly necessary to provide the answer. Moreover, any information that is deemed irrelevant by the Language module never reaches the next module. These two properties reflect the principles of data minimization and purpose limitation, respectively.

This architecture also fits a client-server scenario where the Language module is hosted on the client-side while the Argumentation module is hosted on the server-side. The client can be implemented as an application to be installed on the personal device of the user, while the sessions of interaction with the server are completely anonymous: the personal information of the user never leaves their device unless it is relevant for the answer.

We shall remark that our main focus is on the processing of the user's personal data and how it is used to produce the final answer. Other aspects such as the management of the dialogue will have to be addressed when implementing the system, but they are beyond the scope of this work, since they do not pose a challenge to users' privacy

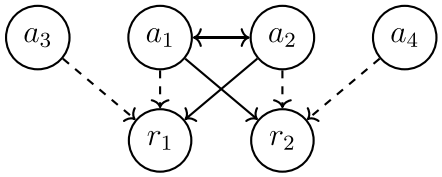


**Fig. 2.** An argumentation graph.

```
1: startConversation()
2: U ← acquireUserSentence()
3: S ← ∅, RY ← ∅, r ← NULL
4: while U is not a stop sentence do
5:    if U is not an explanation request then
6:       N ← computeMatches(U)
7:       S ← S ∪ N
8:       ⟨Cons, PCons⟩ ← retrieveReplies(S, N)
9:       if Cons ≠ ∅ then
10:         r ← selectCandidateReply(Cons)
11:         replyToTheUser(r)
12:         RY = RY ∪ {r}
13:      else
14:         if PCons = ∅ then
15:            terminateConversation()
16:         end if
17:         reply←elicit(S, PCons, RY)
18:         if reply is FALSE then
19:            terminateConversation()
20:         end if
21:      end if
22:   else
23:      Expl ← retrieveExplanation(S, r)
24:      replyToTheUser(Expl)
25:   end if
26:   U ← acquireUserSentence()
27: end while
```

**Algorithm 1.** Dialogue System.

and therefore can be realized with any of the techniques already available in literature.

In the rest of this section, we will describe each component in detail and provide a formal definition of the communication process.

### 3.1. Knowledge base

We encode our background knowledge into an argument graph made of *status* nodes and *reply* nodes. The former encode *facts* that correspond to the possible user sentences. Each status node is linked to one or more *reply* arguments it *endorses*. Status nodes may also attack other status or reply nodes, typically because the facts they represent are incompatible with one another. Indeed, in our argumentation graph, we assume that all the attacks between status arguments are mutual.

**Definition 3.1**. (*Argumentation graph*) An argumentation graph is a tuple $G = \langle A,R,D,E\rangle$, where $A$ and $R$ are the arguments of the graph and are called *status* arguments and *reply* arguments, respectively, $D \subseteq A \times (A \cup R)$ encodes the attack relation and it is such that for each $(a,b) \in D \mid a,b \in A$ it holds that also $(b,a) \in D$, and $E \subseteq A \times R$ encodes the

```
1: for all r ∈ PCons do
2:    N* ← selectDefenceNodes(S, r)
3:    N^new ← ∅
4:    for all n ∈ N* do
5:       N=replyAcquireMatch(n)
6:       N^new ← N^new ∪ N
7:    end for
8:    S ← S ∪ N^new
9:    if N^new ⊆ S then
10:      if RY ≠ ∅ then
11:         r =getReply(RY)
12:         replyToTheUser(r)
13:         return TRUE
14:      end if
15:   else
16:      if r is a consistent reply then
17:         replyToTheUser(r)
18:         RY = RY ∪ {r}
19:         return TRUE
20:      else
21:         ⟨Cons, PCons′⟩ ← retrieveReplies(S, N^new)
22:         if Cons ≠ ∅ then
23:            r ← selectCandidateReply(Cons)
24:            replyToTheUser(r)
25:            RY = RY ∪ {r}
26:            return TRUE
27:         else
28:            PCons = PCons ∪ PCons′
29:         end if
30:      end if
31:   end if
32: end for
33: return FALSE
```

**Algorithm 2.** Function elicit($S, PCons, RY$).

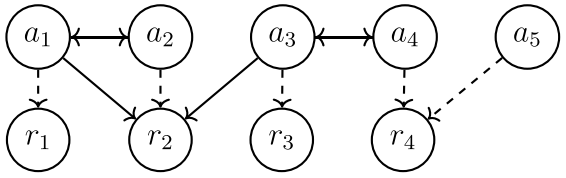


**Fig. 3.** The argumentation graph $G_1$.

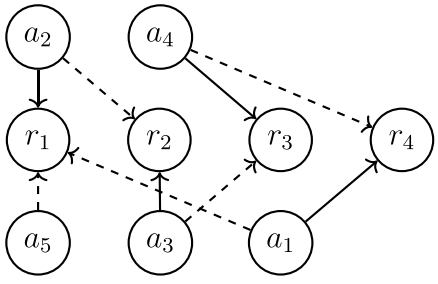


**Fig. 4.** A not well-formed argumentation graph.

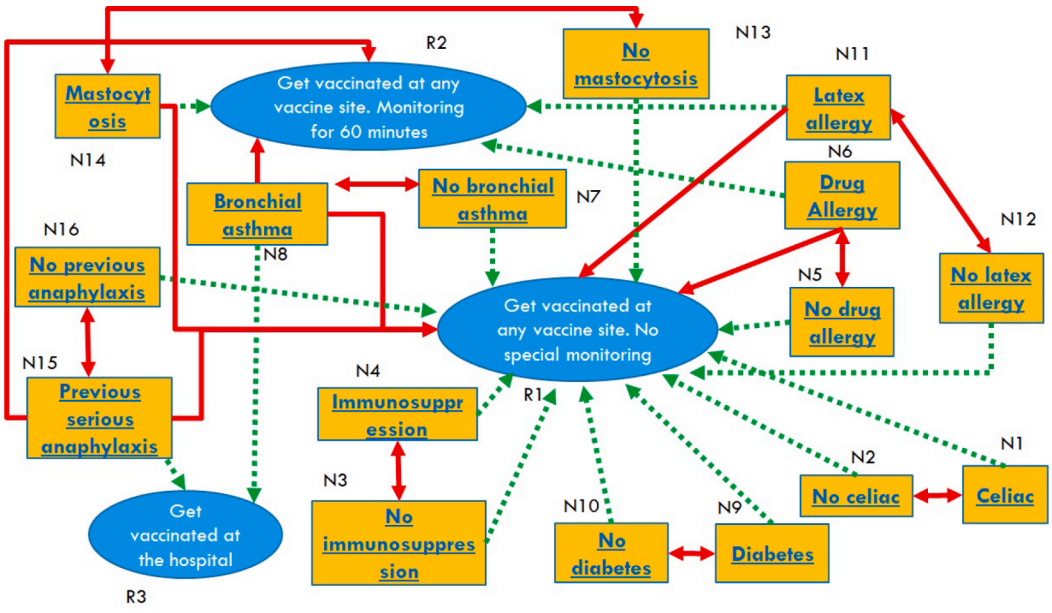


**Fig. 5.** An excerpt of an argumentation graph encoding knowledge about COVID-19 vaccines.

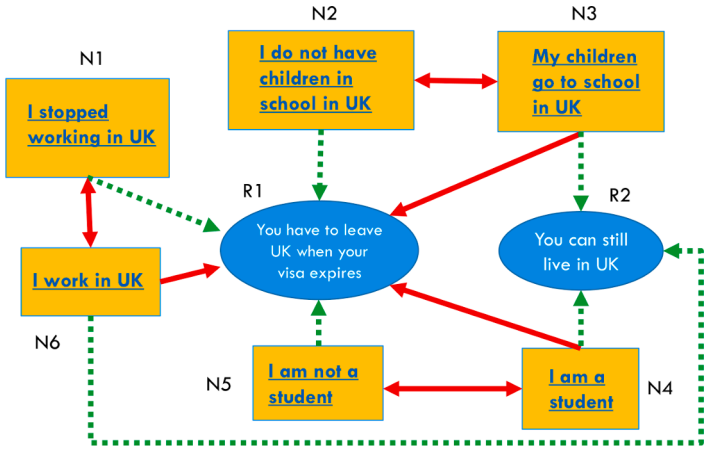


**Fig. 6.** An excerpt of an argumentation graph encoding knowledge about immigration in the UK.

endorsement relation[7].

We say that $a$ *attacks* (resp., *endorses*) a reply node $r$ iff $(a,r) \in D$ (resp., $(a,r) \in E$). By extension, we say that a set $S$ attacks (resp., endorses) $r$, or equivalently that $r$ is attacked by (resp., endorsed by) $S$, iff there exists an argument $a \in S$ s.t. $a$ attacks (resp., endorses) $r$.

**Example 1.** Consider the argumentation graph $G = \langle A, R, D, E \rangle$ in Fig. 2, where $A = \{a_1, a_2, a_3, a_4\}$, $R = \{r_1, r_2\}$, $D = \{(a_1, a_2), (a_2, a_1), (a_1, r_2), (a_2, r_1)\}$, $E = \{(a_1, r_1), (a_2, r_2), (a_3, r_1), (a_4, r_2)\}$, where dashed lines denote the endorsement relation and continuous lines denote the attack relation. Among the several sets attacking or endorsing replies, we have that: set $S_1 = \{a_1, a_2\}$ attacks both $r_1$ and $r_2$, while set $S_2 = \{a_1, a_3\}$ attacks $r_2$, and set $S_3 = \{a_3, a_4\}$ endorses both $r_1$ and $r_2$.

Additionally, each argument in $A$ is annotated with a set of natural language sentences, that represent some possible ways a user would express the fact $A$ is meant to encode. These different representations of facts could be produced by domain experts or crowd-sourced as proposed by Chalaguine and Hunter (2020) and then validated by domain experts. Each argument in $R$ is annotated with one or multiple natural language sentences that express the answer it encodes.

### 3.2. Language module

The Language module has a double purpose: to map the user information with the proper KB nodes, and to filter out sensitive information.

Similarly to Chalaguine and Hunter (2020), we aim to compare the information provided by the user with the natural language sentences in the KB in particular those associated with status nodes. Once we have determined which of these KB sentences are similar to the input sentences, we obtain a set $S$ of associated status nodes. Such a set represents all the information communicated by the user that is relevant for the

[7] Note that our endorsement relation is different from the support relation defined in Bipolar AFs (BAFs) by Cayrol and Lagasquie-Schiex (2005) and Boella et al. (2010). In fact, BAFs have a unique set of arguments $A$ and the support relation is a subset of $A \times A$, instead our endorsement relation is a subset of $A \times R$, thus it only involves pairs $\langle status, reply \rangle$. Furthermore, the support relation of BAFs also affects the extensions (in fact, several variants of the admissible semantics have been defined depending on the type of considered support), instead, our endorsement relation only affects the choice of the replies to be given to the user: the fact that a node is accepted w.r.t. a set only depends on the attack relation, as done in the classical AFs.

**Table 1**
Preliminary experimental results of the embedding models and the threshold criterion on the sentence matching task. For each model, we report only 3 fixed thresholds, the ones that have reached best precision, recall, and F1.

| Model | Threshold | P | R | F1 |
|---|---|---|---|---|
| | mean (0.19) | 0.27 | 0.71 | 0.39 |
| | mean+std (0.41) | 0.34 | 0.39 | 0.36 |
| TF-IDF | 0.65 | 0.50 | 0.14 | 0.22 |
| | 0.25 | 0.28 | 0.61 | 0.38 |
| | 0.20 | 0.26 | 0.68 | 0.38 |
| | mean (0.46) | 0.33 | **1.00** | 0.50 |
| | mean+std (0.63) | 0.99 | **1.00** | **0.99** |
| MPNet-S | 0.70 | **1.00** | 0.86 | 0.92 |
| | 0.65 | **1.00** | 0.97 | **0.99** |
| | 0.60 | 0.92 | **1.00** | 0.96 |
| | mean (0.51) | 0.32 | **1.00** | 0.49 |
| | mean+std (0.67) | 0.99 | **1.00** | **0.99** |
| MPNet-P | 0.75 | **1.00** | 0.86 | 0.92 |
| | 0.70 | **1.00** | 0.94 | 0.97 |
| | 0.65 | 0.96 | **1.00** | 0.98 |
| | mean (0.52) | 0.31 | **1.00** | 0.47 |
| | mean+std (0.68) | 0.99 | 0.97 | 0.98 |
| MPNet-PM | 0.75 | **1.00** | 0.81 | 0.90 |
| | 0.70 | 0.98 | 0.93 | 0.96 |
| | 0.65 | 0.90 | **1.00** | 0.95 |
| | mean (0.46) | 0.40 | **1.00** | 0.57 |
| | mean+std (0.62) | 0.72 | 0.99 | 0.83 |
| TBERT-P | 0.75 | **1.00** | 0.46 | 0.63 |
| | 0.65 | 0.81 | 0.94 | 0.87 |
| | 0.60 | 0.61 | **1.00** | 0.75 |
| | mean (0.42) | 0.43 | **1.00** | 0.60 |
| | mean+std (0.63) | 0.55 | 0.96 | 0.70 |
| MiniLM-P | 0.75 | 0.81 | 0.43 | 0.57 |
| | 0.65 | 0.57 | 0.87 | 0.69 |
| | 0.60 | 0.51 | **1.00** | 0.68 |
| | mean (0.39) | 0.37 | **1.00** | 0.54 |
| | mean+std (0.59) | 0.50 | 0.75 | 0.60 |
| DBERT-NQ | 0.80 | **1.00** | 0.17 | 0.30 |
| | 0.55 | 0.49 | 0.86 | 0.62 |
| | 0.40 | 0.39 | **1.00** | 0.56 |

**Table 2**
Nodes used in our case study and example of sentences associated with them.

| Node ID | Sent. ID | Sentence |
|---|---|---|
| N1 | S1 | I am celiac |
| N2 | S8 | I am not celiac |
| N3 | S12 | I do not suffer from immunosuppression |
| N4 | S19 | I recently found out to be immunosuppressed |
| N5 | S20 | I do not have any drug allergy |
| N6 | S28 | I have a serious drug allergy |
| N7 | S34 | I've never had bronchial asthma |
| N8 | S38 | I am affected by bronchial asthma |
| N9 | S41 | I am diabetic |
| N10 | S46 | I don't have diabetes |
| N11 | S50 | Latex causes me an allergic reaction |
| N12 | S52 | I not allergic to latex |
| N13 | S59 | Mastocystosis is not an health concern for me |
| N14 | S60 | I suffer from mastocytosis |
| N15 | S66 | I went into anaphylactic shock before |
| N16 | S67 | I've never experienced a serious anaphylaxis |

task at hand, therefore the information that is needed for the Argumentation module to compute the answer.

This representation is completely anonymous and general since it does not include the original inputs of the user, nor information regarding the single matched sentences. It can be effectively considered a sanitized version of the input of the user. Also, any additional irrelevant, but potentially sensitive, information that the user has provided will result dissimilar from the KB sentences. Therefore, $S$ is the minimal information, in terms of quantity and format, among that provided by the user that is necessary to compute the answer.

There are many possible strategies that can be used to compute the match, with no consequences on the rest of the architecture as long as they are accurate. Chalaguine and Hunter (2020, 2021) represent sentences using TF-IDF vectors and compare them using cosine similarity (Kenter and de Rijke, 2015), selecting only the most similar sentences in their KB. We instead propose to use sentence embeddings, which allow representing a sentence as a real-valued vector, mapping it into a semantic vector space. As opposed to TF-IDF, this technique allows mapping sentences with similar semantic content into nearby vectors, even if they are very different from a lexical point of view. Instead of cosine similarity, we propose to use Bray-Curtis similarity (Bray and Curtis, 1957), since it has led to satisfactory results in the context of sentence similarity (Galassi et al., 2020). Finally, since the user's sentence may contain information related to multiple nodes of the KB, we do not select only the most similar node, but instead, use a threshold hyper-parameter to discriminate between similar and dissimilar sentences.

Among the many possible sentence embeddings, we have decided to use Sentence-BERT (Reimers and Gurevych, 2019), which are based on deep network models trained on a vast amount of data and designed specifically for the task of sentence similarity. This choice is motivated by the key role they have played in the advancement of Natural Language Processing in recent years. The choice of whether to train an embedding model from scratch, use a pre-trained one, or fine-tune a pre-trained one depends on the specific domain of application and the data that may be available for training. The downside of this approach is that it requires hardware resources capable of loading the neural model, which may be unfeasible in some contexts. As an alternative, GloVe embeddings (Pennington et al., 2014) are usually less performing, but do not involve the use of neural models and therefore may be applicable in the general case.

We have proposed to compute matches by similarity between sentence embeddings, but it is important to remark that our general architecture would be compatible also with other methods. A possible alternative would be to use techniques that directly compute the similarity of the two sentences. This could be implemented either using specific algorithms such as the Damerau-Levenshtein dissimilarity (Damerau, 1964), or neural networks such as Poly-Encoders (Humeau et al., 2020). However, this alternative would have a heavy computational footprint, since it would require processing every pair of sentences at run-time. As opposite, the approaches based on sentence embeddings would be very fast, since all the KB embeddings can be pre-computed and the comparison between numerical vectors is rather inexpensive. Finally, it is important to be aware that the ability of sentence embedding to model some concepts may still be imperfect. For example, BERT (Devlin et al., 2019) may better capture negations (Lin et al., 2020; Zhu et al., 2018), whereas GloVe may better understand punctuation (Karami et al., 2021). In some cases, it may be necessary to partially or completely rely on other techniques so as to obtain better matches, e.g., additional pre-processing steps, stance detection, rule-based models, or the use of ontologies.

### 3.3. Argumentation module

The Argumentation Module is in charge of computing the replies to the user queries/sentences. To appreciate our approach, it is important to understand the limitations of choosing a reply only based on the last user sentence, as done by Chalaguine and Hunter (2020). Let us consider the case where a user interacts with a dialogue system in the context of the vaccines for COVID-19, to understand whether they can get safely vaccinated. In this case, there can be conditions making some candidate replies invalid that are not revealed or excluded by the content of the user last sentence, and further information is required: this information can be already available (as it is contained in what the user has previously declared) or needs to be collected, by gathering the users answers to specific new questions.

For instance, suppose that the user says: æI am celiac, can I get vaccinated?ɡ. In the case of celiac people that suffer from no other

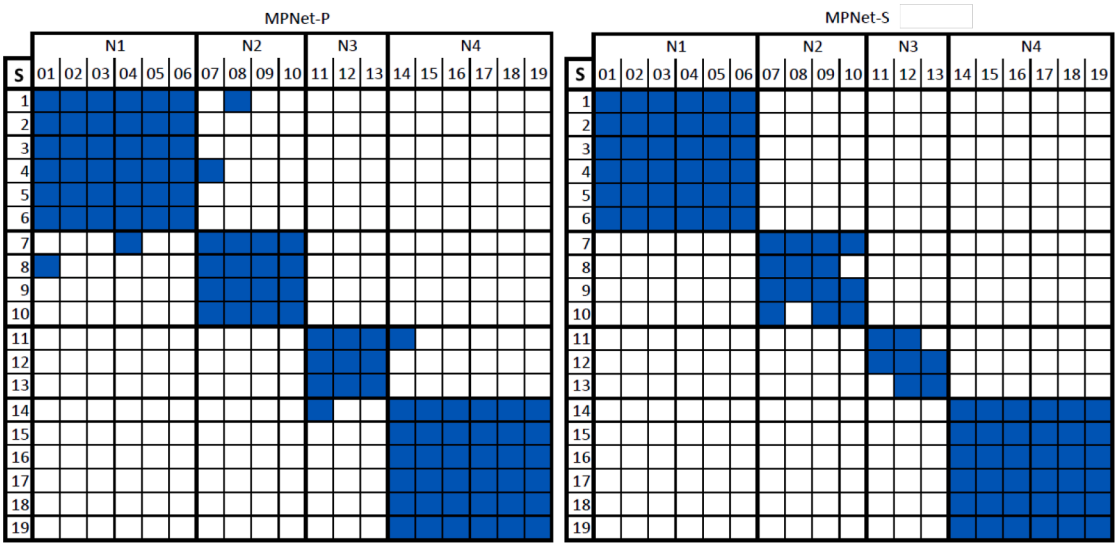


**Fig. 7.** Matches computed by the models using the 0.65 threshold value on sentences from S1 to S19. The colored cells indicate the matches computed by the two models.

disease, the answer to this question is $R$ = æYes, you can get vaccinated at any vaccine site. No special monitoring time is neededg.[8] In fact, there is no known specific side effect for people suffering from celiac disease, thus there is no need for celiac people to get vaccinated in the hospital or to undergo a specific monitoring time. However, suppose that the user also suffers from bronchial asthma. In this case, the answer that correctly follows the AIFA guidelines is æYes, but you must get vaccinated at the hospitalg.

The point is that choosing how to reply to the user by only looking at the current user sentence may be unsafe: the dialogue system should further investigate the health conditions of the user in order to exclude any pathology that makes a candidate reply inappropriate. In the example above, before giving the reply $R$ to the user, the dialogue system should ask the user whether they suffer from bronchial asthma and/or from the other (few) diseases that make $R$ inappropriate. Furthermore, the dialogue system should keep track of everything the user said (differently from what done by Chalaguine and Hunter), because the reply to any further users question must be given by taking into account all the relevant information provided by the user, otherwise there is the risk of giving a reply that could mislead the user.

### 3.3.1. Reply strategy

We are ready to present our strategy for providing users with replies and the algorithm encoding it. Each dialogue session relies on *dynamically acquired knowledge*, expressed as a set of status arguments $S$, that encode user information. Basically, $S$ contains the status nodes of the KB *activated* so far, that is corresponding to the information the user has communicated to the system since the starting of the dialogue session.

Differently from other proposals, at each turn, our system does not simply select a reply endorsed by $S$. On the contrary, the aim of the dialogue strategy is to provide the user with a reply that is both endorsed and defended by $S$. In other words, the system works to provide only robust replies, possibly delaying replies that need further fact-checking.

In fact, our system distinguishes between *consistent* and *potentially consistent* reply. The former can be given to the user right away, as, as formally stated in Section 3.3.2, it can not possibly be proven wrong in the future.[9] The latter, albeit consistent with the current known facts, may still be defeated by future user input, and therefore it should be delayed until a successful elicitation process is completed. The formal definitions, reported below, are based on the KB and on a representation of the state of the dialogue consisting of a set $S$. In particular, $S \subseteq A$ contains all the arguments activated during the conversation so far and is assumed to be conflict-free. Furthermore, both definitions are based on the concept of *acceptable* arguments, recalled in Section 2.

**Definition 3.2.** (*Consistent reply*) Given an argumentation graph $G = \langle A, R, D, E\rangle$ and a conflict-free set $S \subseteq A$, a reply $r \in R$ is *consistent* w.r.t. $S$ iff $S$ endorses $r$ and $r$ is acceptable w.r.t. $S$.

**Definition 3.3.** (*Potentially consistent reply*) Given an argumentation graph $G = \langle A, R, D, E\rangle$ and a conflict-free set $S \subseteq A$, a reply $r \in R$ is *potentially consistent* w.r.t. $S$ iff $S$ endorses $r$, $S$ does not attack $r$ and $r$ is not acceptable w.r.t. $S$.

As it will be clearer in the next section, the aim of the strategy is that of looking for consistent replies to be given in response to the user input. If no consistent reply exists, then the strategy is that of looking for potentially consistent replies and trying to make one of them a consistent reply. The following example shows consistent and potentially consistent replies.

**Example 2.** Consider the argumentation graph $G = \langle A, R, D, E\rangle$ introduced in Example 1, where $A = \{a_1, a_2, a_3, a_4\}$, $R = \{r_1, r_2\}$, $D = \{(a_1, a_2), (a_2, a_1), (a_1, r_2), (a_2, r_1)\}$, $E = \{(a_1, r_1), (a_2, r_2), (a_3, r_1), (a_4, r_2)\}$, and $S = \{a_3\}$. Because $r_1$ is attacked by $a_2$ and it is not defended by $S$, $r_1$ is not a consistent reply. However, $r_1$ is a potentially consistent reply as it is endorsed by $S$ and not attacked by $S$. In the case that $S = \{a_1, a_3\}$, instead, $r_1$ is a consistent reply.

In addition to provide replies, the system is also able to provide ex-

[8] Taken from the FAQ section of the AIFA web site: https://www.aifa.gov.it/en/vaccini-covid-19/.

[9] The implicit assumption here is that the user does not enter conflicting information, and that the language model correctly interprets the user input. Clearly, if this is not the case, the system's output becomes unreliable. But that wouldn't depend on the underlying reasoning framework. The definition of fallback strategies able to handle such exceptions would be an important extension to the system.

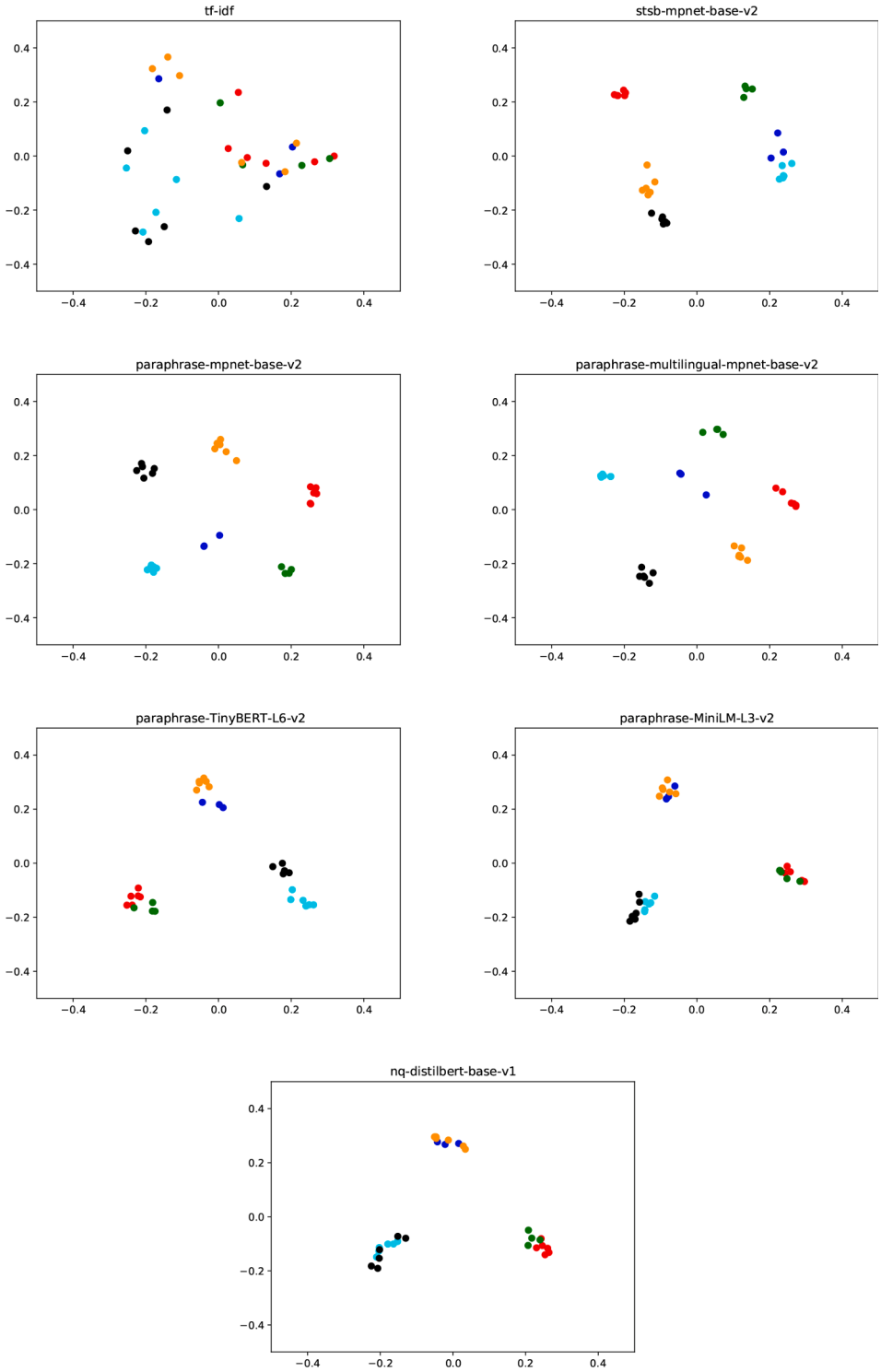


**Fig. 8.** Visualization of the encoded sentences after PCA projection and normalization. Nodes from N1 to N6 are represent through different colors; in order: red, green, blue, orange, cyan, and black. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

planations for the given replies. An *explanation* of a reply $r$ consists of two parts. The first one contains the arguments leading to $r$, i.e., those belonging to a set $S$ that endorses $r$. The second one encodes the *why nots*, to explain why the system did not give other replies.

**Definition 3.4.** (*Explanation*) Given an argumentation graph $G = \langle A, R, D, E \rangle$, a set $S \subseteq A$ and a reply $r \in R$, an *explanation* for $r$ is a pair $\langle End, NotGiven \rangle$, where *End* contains the arguments $a \in S$ s.t. $(a, r) \in E$ and *NotGiven* is a set of pairs $\langle r', N' \rangle$, where $r' \neq r$, $r'$ is endorsed by $S$ and $N' \subseteq S$ contains the arguments $b$ attacking $r'$.

**Example 3.** Continuing the previous example, an explanation for $r_1$ in the case that $S = \{a_1, a_3, a_4\}$, is given by $\langle \{a_1, a_3\}, \{\langle r_2, \{a_1\}\rangle\}\rangle$, meaning that $r_1$ has been given to the user since it is endorsed by both $a_1$ and $a_3$, and that $r_2$ could not have been given to the user (although it is endorsed by $S$) since it is attacked by $a_1$.

The behaviour of our dialogue system is specified by Algorithm 1. Initially, the system starts the conversation with the user (line). This

**Table 3**
Experimental results of the selected embedding models and the threshold criterion on the sentence matching task.

| Model | Threshold | P | R | F1 |
|---|---|---|---|---|
| | mean+std (0.63) | 0.98 | 0.87 | 0.92 |
| | 0.70 | **1.00** | 0.60 | 0.75 |
| MPNet-S | 0.65 | **1.00** | 0.82 | 0.90 |
| | 0.60 | 0.95 | **0.98** | **0.96** |
| | mean+std (0.67) | 0.92 | 0.89 | 0.91 |
| | 0.75 | **1.00** | 0.53 | 0.69 |
| MPNet-P | 0.70 | **1.00** | 0.76 | 0.87 |
| | 0.65 | 0.88 | 0.91 | 0.89 |
| | mean+std (0.68) | **1.00** | 0.85 | 0.92 |
| | 0.75 | **1.00** | 0.64 | 0.78 |
| MPNet-PM | 0.70 | **1.00** | 0.85 | 0.92 |
| | 0.65 | 0.95 | 0.96 | 0.95 |

where *Cons* is a set of consistent replies, according to Definition 3.2. Instead, set *PCons* contains the potentially consistent replies, that are reply arguments endorsed by $S$ but are not acceptable in $S$ at the moment, as per Definition 3.3. This basically means that an argument $a \in PCons$ could turn to be acceptable in $S$ by adding some new argument to $S$ making $S$ defend $a$, and this is done by collecting more information by the user.

Then the operations aimed at finding a reply to be given to the user start. If *Cons* is not empty, a reply is arbitrarily selected among those in *Cons* (line), stored in $RY$ and returned to the user ().[10] In case both *Cons* and *PCons* are empty (line), a consistent reply can not be found and the conversation is terminated. Otherwise, if *PCons* is not empty, Algorithm 1 starts the elicitation strategy, aimed to turn some reply in *PCons* consistent. Specifically, function elicit is invoked (line), that receives as input sets $S$, *PCons* and $RY$ and returns a boolean whose value indicates

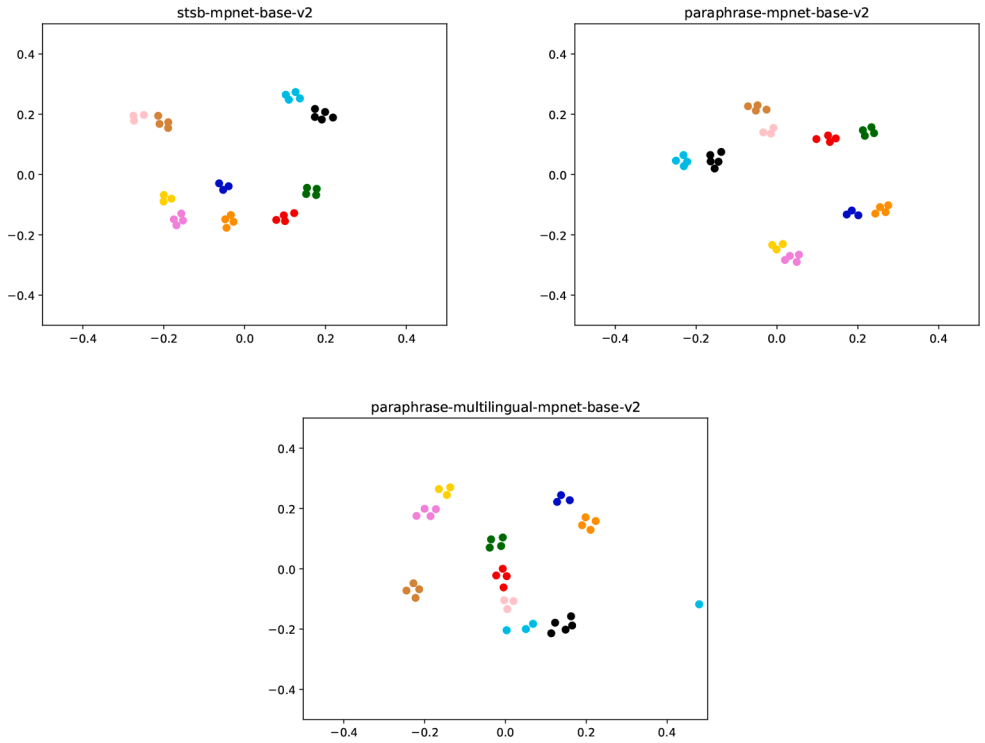


**Fig. 9.** Visualization of the encoded sentences after PCA projection and normalization. Nodes from N7 to N16 are represent through different colors; in order: red, green, blue, orange, cyan, black, violet, yellow, brown, and pink. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

includes understanding the user question and the context of reasoning. In this work we do not focus on how this method is implemented, but on how to collect the relevant information and how to provide the correct reply. At line, the first user sentence is acquired and stored into variable $U$. Line initializes the set $S$ that will be used to store the arguments activated during the conversation, set $RY$ that will be used to store the replies given to the user and variable $r$ that will be used to store the current reply to be given to the user.

The **while** loop (line) handles the conversation with the user, until they terminate the chat by using a closing sentence. In the case that $U$ is not an explanation request (), function computeMatches is invoked (line), whose task is performed by the language module and consists in matching the relevant information given by the user with the status arguments of the KB. The output of function computeMatches is a set $N$ of status arguments, that are first added to $S$ (line) and then given as input to function retrieveReplies in order to retrieve the reply arguments that are endorsed by $S$, and in particular by $N$, that contains the last activated nodes. In particular, the output of retrieveReplies is a pair $\langle Cons, PCons \rangle$, the outcome of the elicitation process: if its value is TRUE it means that a consistent reply has been found and given to the user by the function, and that sets $S$ and $RY$ have been correctly updated, thus the while loop continues the conversation with the user by acquiring a new user sentence (line), otherwise it means that no reply in *PCons* turned out to be consistent and that no new consistent reply has been found, thus the conversation must be terminated (line). More detail on function elicit will be given shortly. In the case that $U$ is an explanation request (line), meaning that the user is looking for an explanation for the last reply $r$ the system gave to them, the proper explanation according to Definition 3.4, is retrieved at line, and given to the user (line).

Function elicit works as follows. Every potential consistent reply $r$ belonging to *PCons* is examined by the for loop at line, then the arguments not belonging to $S$ that attack the attackers of $r$ are retrieved

[10] This selection could be made by a more sophisticated strategy, but we leave this to future work.

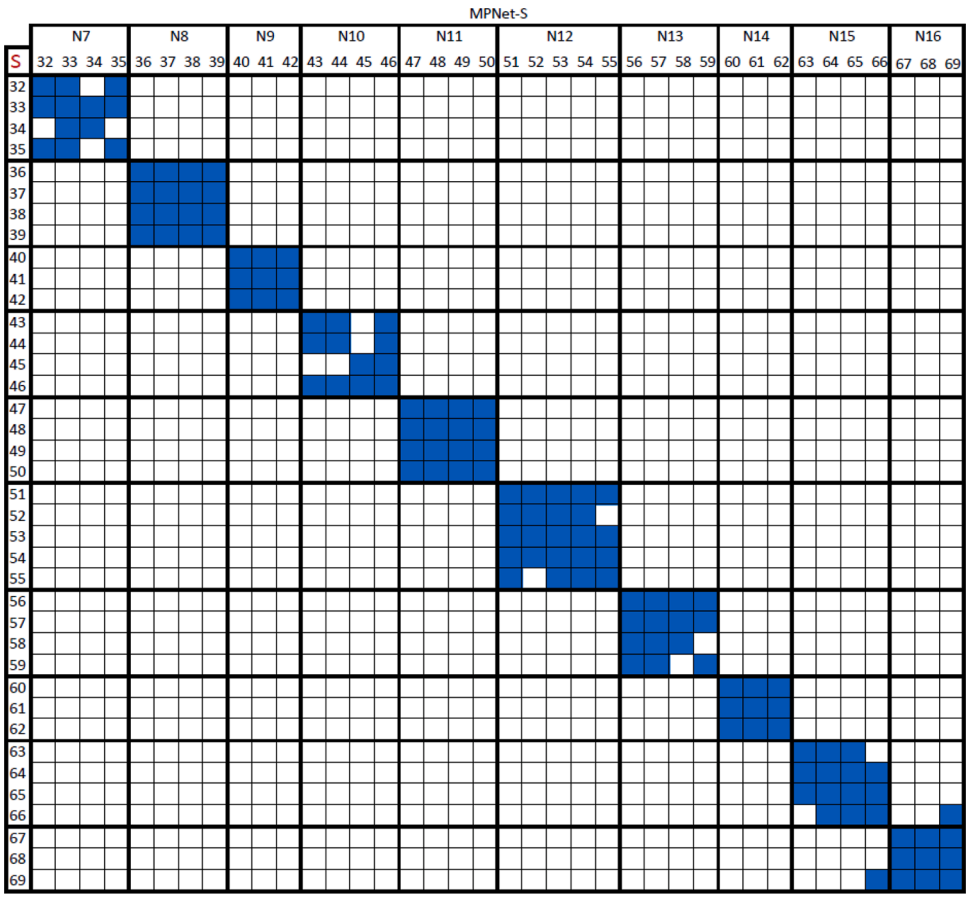


**Fig. 10.** Matches computed by MPNet-S using the mean+std threshold value on the test set.

(line). Each of these arguments $n$ (line) is transformed in a proper sentence and submitted to the user (line), to see if the user confirms or denies the information contained in $n$. The reply of the user to each $n$ is collected and the arguments activated by the reply are added to set $N^{new}$ (line). At the end of this inner for loop, if $N^{new}$ is a empty or a subset of $S$, $r$ is still not consistent, then the only operations that the algorithm can do is giving a reply belonging to $RY$ and returning TRUE (line). In the case that $N^{new}$ contains new arguments that make $r$ to be a consistent reply (line), $r$ is given to the user, $r$ is added to $RY$ and the function returns TRUE. In the case that $r$ is still not consistent, instead, new candidate replies are retrieved (line) and then (i)if *Cons* is not empty, one reply is selected and given to the user (line), $r$ is added to $RY$ and TRUE is returned, otherwise (ii) *PCons* is updated () and the main for loop continues with another iteration.

As regards the worst-case complexity of our algorithms, it is easy to see that each iteration of the while loop of Algorithm 1 can be executed in time $O(H \times \max(K,H))$, where $H$ is the number of status nodes and $K$ is the number of reply nodes. In fact, both functions computeMatches and retrieveReplies are $O(H)$, while selectCandidateReply is $O(K)$, retrieveExplanation is $O(H \times K)$, and elicit is $O(H \times \max(K,H))$. In particular, the complexity of elicit is determined by the complexity of the main loop ($O(K)$) multiplied by the complexity of most expensive operation, that can be the for loop at line ($O(H)$), or one between selectCandidateReply and retrieveReplies. It should be noted that, in practice, all the operations require much less time than the theoretical upper bound, due to the usage of indexes and suitable data structures.

Section 4 provides an example of how Algorithm 1 works.

### 3.3.2. Properties

Our approach enjoys some interesting properties. The first one is a property of consistent replies. Indeed, the fact that a reply $r$ is consistent means that $S$ counterattacks every attack towards $r$, thus as the algorithm proceeds and $S$ grows, no status arguments added to $S$ can make $r$ inconsistent, as long as $S$ remains conflict-free, i.e., as long as the user does not make conflicting statements.

**Proposition 1.** *Given an argumentation graph $\langle A,R,D,E\rangle$ and a set $S \subseteq A$, a consistent reply $r$ w.r.t $S$ is a consistent reply for any conflict-free set $S' \supseteq S$.*

**Proof.** We prove the statement reasoning by contradiction. Suppose that $S' \supseteq S$ is conflict free and $r$ is not a consistent reply for $S'$. This means that (i)$S'$ does not endorse $r$ or (ii)$N \in S' \setminus S$ attacks $r$. As regards (i), since $S$ is a subset of $S'$, the fact that $S'$ does not endorse $r$ contradicts the hypothesis that $r$ is a consistent reply w.r.t $S$. As regards (ii), since $r$ is acceptable w.r.t. $S$, then $S$ attacks all the arguments attacking $r$, meaning that $S$ also attacks $N$, contradicting the fact that $S'$ is conflict-free.□

**Example 4.** Consider the argumentation graph $G_1 = \langle A,R,D,E\rangle$ depicted in Fig. 3, where $A = \{a_1,a_2,a_3,a_4,a_5\}$, $R = \{r_1,r_2,r_3,r_4\}$, $D = \{(a_1,a_2),(a_2,a_1),(a_1,r_2),(a_3,r_2),(a_3,a_4),(a_4,a_3)\}$, $E = \{(a_1,r_1),(a_2,r_2),(a_3,r_3),(a_4,r_4),(a_5,r_4)\}$. It is easy to see that $S = \{a_2\}$ has no consistent reply, while $S' = \{a_2,a_4\}$ has two consistent replies that are $r_2$ and $r_4$, and that no conflict-free superset of $S'$ exists making $r_2$ not a consistent reply w.r.t. it.

Now we introduce the concepts of *inconsistent* set and *well-formed* argumentation graph, that will be used to state the existence of potential consistent and/or consistent replies.

**Definition 3.5.** (*Inconsistent set*) Given an argumentation graph $\langle A,R,D,E\rangle$, a set $K \subseteq A$ is an *inconsistent* set of arguments if and only if it is conflict-free and every $r \in R$ that is endorsed by $K$ is also attacked by $K$.

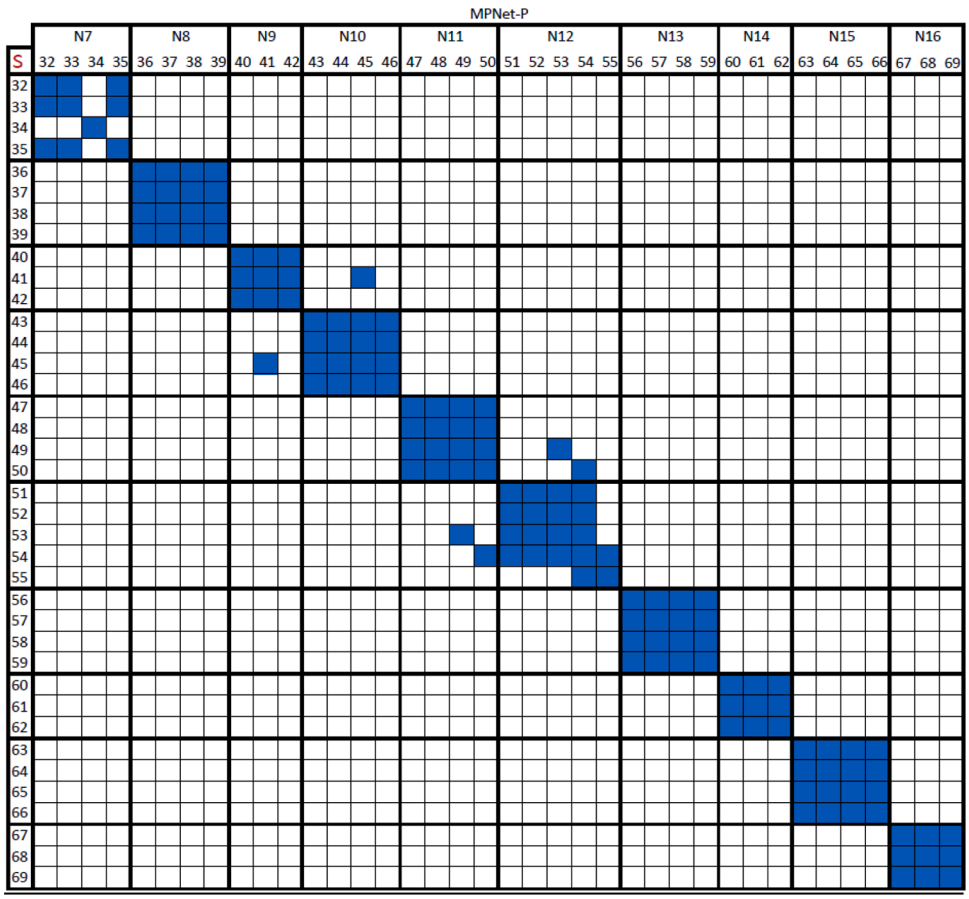


**Fig. 11.** Matches computed by MPNet-P using the mean+std threshold value on the test set.

Basically, an inconsistent set is a set that admits no potential consistent replies, and thus no consistent replies at all. In the case that no inconsistent set exists in the argumentation graph $G$, $G$ is said to be *well-formed*.

**Definition 3.6.** (*Well-formed Argumentation Graph*) An argumentation graph $\langle A, R, D, E\rangle$ is *well-formed* if and only if there not exists any inconsistent set $K \subseteq A$.

**Example 5.** Consider the argumentation graph $G_2 = \langle A, R, D, E\rangle$ depicted in Fig. 4, where $A = \{a_1, a_2, a_3, a_4, a_5\}$, $R = \{r_1, r_2, r_3, r_4\}$, $D = \{(a_1, r_4), (a_2, r_1), (a_3, r_2), (a_4, r_3)\}$, $E = \{(a_1, r_1), (a_2, r_2), (a_3, r_3), (a_4, r_4), (a_5, r_1)\}$. It is easy to see that $K = \{a_1, a_2, a_3, a_4\}$ is an inconsistent set, making $G_2$ a not well-formed argumentation graph.

The following property concerns the replies provided by our Algorithm.

**Proposition 2.** *In the case that the input argumentation graph $G = \langle A, R, D, E\rangle$ is well-formed and $G$ is such that $\forall a \in A$ there exists $r \in R$ s.t. $(a, r) \in E$, the output of function* retrieveReplies *is such that $\langle Cons, PCons\rangle \neq \langle \emptyset, \emptyset\rangle$ at each invocation.*

**Proof.** Reasoning by contradiction, assume that $\langle Cons, PCons\rangle = \langle \emptyset, \emptyset\rangle$ for some $S$. The fact that both *Cons* and *PCons* are $\emptyset$ means that every $r \in R$ is such that $r$ is not endorsed by $S$ or is attacked by $S$, otherwise at least one between *Cons* and *PCons* would be different from $\emptyset$. The case that no $r \in R$ is endorsed by $S$ contradicts the hypothesis that $\forall a \in A$ there exists $r \in R$ s.t. $(a, r) \in E$, thus it must be the case that every $r$ endorsed by $S$ is also attacked, but this contradicts the hypothesis that $G$ is well formed.□

The property reported above means that, in the case that the graph is well-formed, at least one of the sets outputted by the function is different from the empty set, meaning, in turn, that there is at least one potential consistent reply or one consistent reply at each iteration or both.

The following property regards the termination of Algorithm 1.

**Proposition 3.** *Function* elicit *terminates.*

**Proof.** We prove the statement by examining the alternative scenarios that can occur and showing that the termination is reached in every scenario. The function starts with the main for loop at line that selects a reply $r$ belonging to *PCons* and invokes selectDefenceNodes.□

At this point the two following alternative cases can occur. *Case a)*$N^{new}$ is empty (as a possible consequence of $N^* = \emptyset$ or because replyAcquireMatch returns an empty set) or is a non-empty subset of $S$: in this case, if an already given reply exists, it will be selected (line) and given to the user, terminating the function, or another iteration starts; *Case b)*$N^{new}$ is not empty and not a subset of $S$: in this case, if $r$ is now a consistent reply, the function provides $r$ to the user (line) and terminates at line. Otherwise, in the case that $r$ is still not consistent, the function can terminate at line, or continue with another iteration. Even in the case that another iteration starts, the termination is guaranteed as one of the following two cases will finally occur. *Case i)*all the nodes of the graph have been added to $S$: then, $N^*$ at line is empty and this could cause that the function returns TRUE as in the case a) explained above or that no new nodes can be added to *PCons* at line, that in turns means that, after examining all the replies in *PCons*, the for loop ends and the function returns FALSE at line ; *Case ii)*the user sentence are matched to some nodes already in $S$ (line): it is easy to see that this case results in returning TRUE at line or in returning FALSE at line, when all the replies in *PCons* have been examined by the for loop. the case that the input argumentat

Note that, if the user does not contradict himself, $S$ is admissible in

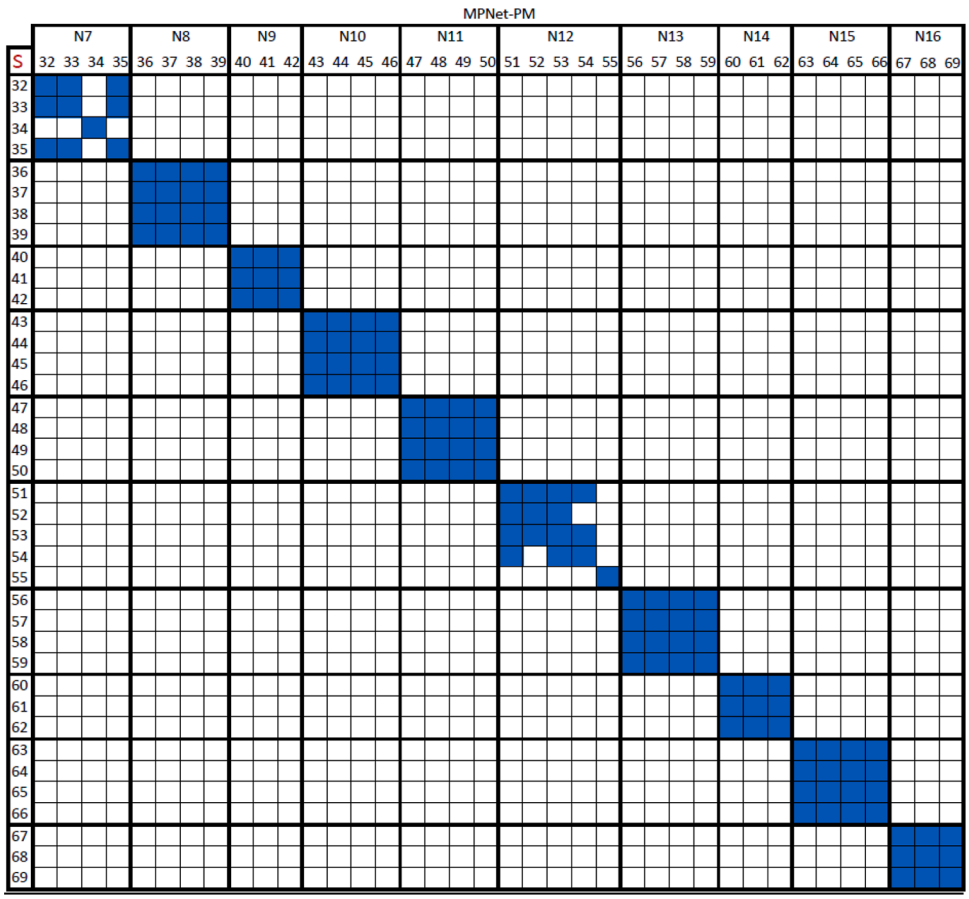


**Fig. 12.** Matches computed by MPNet-PM using the mean+std threshold value on the test set.

every iteration of the algorithm. In fact, since the attacks are mutual, every node added to $S$ counterattacks by itself the attacks it receives, thus every added node is acceptable. This guarantees that the reply given to the user are endorsed and defended by an admissible set of status nodes, enforcing the reasonableness of the replies.

## 4. Case study

In this section, we provide an example of how our algorithm works by showing a case study for the context of the vaccines for COVID-19.

Here the aim of creating a dialogue system able to accurately answer user questions about vaccine modalities and more. A concrete scenario may include a government agency providing the dialogue service to its citizens while relying on a third-party scientific institution (e.g., a research center) for the argumentation service and the knowledge base, or one where citizens can use a mobile-phone app to retrieve information provided by a research center.

In Fig. 5, we show an excerpt of the argumentation graph encoding the knowledge base,[11] in particular the part related to the modalities of getting vaccinated. In this graph, the yellow rectangles represent the status nodes, the blue ovals represent the possible replies, the green dotted arrows encode the endorsement relations, thus point to the possible replies to a given user sentence, and the red ones denote the attack relations, thus encoding the replies that the system must not give to user sentences matched to the nodes attacking them.

It is worthwhile noticing that the graph contains both the positive and negative version of each status argument. This is a key modeling feature in the context at hand, as it enables the system to properly capture and encode all the information provided by the user about their health conditions. Inside each status node we represent the associated natural language sentences.[12]

Let us consider this example: the user sentence acquired at the first iteration of the while loop is *æHi, I am Morgan and I suffer from latex allergy, can I get vaccinated?ɡ* (line). The language module processes the user sentence and compares it against all the sentences provided by the knowledge base, resulting in a single positive match with the sentence 'I have latex allergy' associated with node $N_{11}$, then $N = N_{11}$ at line. At this point, function retrieveReplies returns $\langle \emptyset, \{R_2\} \rangle$, as $R_2$ is the only reply endorsed by $N$. This reply is not a consistent reply, because it is attacked by both $N_8$ and $N_{15}$. It is, however, a potentially consistent reply. Thus, function elicit is invoked, with $S = \{N_{11}\}$, $PCons = \{R_2\}$ and $RY = \emptyset$. Function elicit invokes function selectDefenceNodes, that returns $\{N_7, N_{16}\}$. In fact, to make $R_2$ consistent, $S$ must be augmented with both $N_7$ and $N_{16}$. This means that the user must tell that they do not suffer from bronchial asthma and that they had no previous anaphylaxis. Then, the inner for loop is executed, at the end of which $N^{new} = \{N_7, N_{16}\}$ (line), supposing that the user does not suffer from the mentioned diseases. Since $R_2$ now is a consistent reply w.r.t. $S = \{N_7, N_{11}, N_{16}\}$, it is given to the user at line and the function terminates.

Alternatively, suppose that the user writes that they do suffer from bronchial asthma. In that case, we would have $S = \{N_{11}, N_8, N_{16}\}$, hence $R_2$ would not be a consistent reply. Accordingly, function retrieveReplies is invoked at line of function elicit, with $N^{new} = \{N_8, N_{16}\}$, that returns $\langle$

[11] We base our example on the content of the AIFA website. Since we have no medical expertise, the examples used in this paper are to be considered for the only purpose of illustrating our proposal, and may not reflect the current recommendations on the topic.

[12] For sake of simplicity, we consider only one sentence for each node.

$\{R_3\}, \varnothing\rangle$. In this case, $R_3$ is given to the user at line and the function terminates.

Besides Covid-19 vaccine information, our architecture can accommodate many other scenarios where privacy matters. In particular it would be most useful in any context where (i) the desired information are publicly available but may be difficult to obtain or to navigate, and (ii) to provide the correct answer it is necessary to know the user's sensitive information. In particular, the motivation for (i) is that it could be possible for a user to reconstruct the argumentative graph of the KB through the use of multiple queries. Therefore our proposal is not suited for scenarios where the reasoning process or the knowledge base must remain hidden. One possible domain of application could be the access to legal information, for example in the context of immigration (Queudot et al., 2020). Fig. 6 shows an example of the KB that can be used to address the problem of whether a immigrant is required to leave UK as soon as they stop having a job in UK.

## 5. Evaluation and discussion

Since the Argumentation module is a symbolic module for knowledge representation and reasoning, its evaluation was based on formal properties such as consistency, well-formedness, and termination (Section 3.3.2). To assess the effectiveness of our Language module instead we run an experimentation on a use case of vaccines for COVID-19. Following the KB illustrated in Fig. 5, we build a small-sized dataset of sentences that are representative of its arguments (i.e., the nodes N1 to N16 of the graph). We are especially interested in evaluating our method on sentences with a similar syntactic structure, but different meaning (e.g., a sentence and its negation). Initially, we consider only 6 argumentative nodes in a preliminary experiment aimed to find the best models and their optimal hyper-parameters. Then, we test our choices on the remaining 10 argumentative nodes. For each node, our KB contains between 3 and 7 natural language sentences that can be used to express that concept (see Table 2 and Appendix A).

To obtain a quantitative evaluation, we frame the task as binary classification of every pair of different sentences that are in our KB. A pair is considered a positive instance if two sentences are associated with the same node, a negative instance otherwise. For each combination of models and thresholds, we measure precision, recall, and F1 score of the positive class. Precision is especially important: false positives can be seen as cases where the system "misunderstands" the input of the user, and therefore precision can be seen as a measure of *correctness*. Recall instead can be seen as a measure of the ability of the system to not "miss" information contributed by the user. For our system, poor recall is a less serious problem than poor precision for two reasons. First, it is necessary to match only one sentence associated with an argumentative node to activate it. Second, the argumentative reasoning module proactively asks the user for missing bits of information that would influence the final result. In our perspective, the priority must be to guarantee the correctness of the final answer, even if this means that the system will, in some cases, ask for information that the user has already submitted. For this reason, we use precision as the main metric of comparison.

### 5.1. Selection of embedding models and threshold values

We consider the TF-IDF representation used by Charras et al. (2016) and Chalaguine and Hunter (2020), along with the following Sentence-BERT models:[13]

- `stsb-mpnet` (MPNet-S): based on MPNet (Song et al., 2020) and pre-trained for semantic similarity on the STSbenchmark (Cer et al., 2017).
- `paraphrase-mpnet` (MPNet-P): based on MPNet and pre-trained for paraphrase mining.
- `paraphrase-multilingual-mpnet` (MPNet-PM): multilingual extension (Reimers and Gurevych, 2020) of the monolingual model. We have decided to include this model in the perspective of future multi-lingual applications.
- `paraphrase-TinyBERT-L6` (TBERT-P): based on TinyBERT (Jiao et al., 2020) and pre-trained for paraphrase mining.
- `paraphrase-MiniLM-L3` (MiniLM-P): based on MiniLM (Wang et al., 2020) and pre-trained for paraphrase mining.
- `nq-distilbert` (DBERT-NQ): based on DistilBERT (Sanh et al., 2019) and pre-trained for question answering on Googles Natural Questions dataset (Kwiatkowski et al., 2019).

In this experiment, we want to investigate a wide range of threshold values, which is the only hyper-parameter of our method. We consider two values that are based on the distribution of the similarity scores: one is given by the average of the similarities (mean), and the other one is given by the sum between the average similarity and the standard deviation (mean+std). Additionally, we consider a set of 13 fixed values ranging from 0.20 to 0.80. Results are shown in Table 1.

Our results clearly show that the MPNet-S and the MPNet-P models are the best ones, with the former achieving perfect precision along with high recall and F1. In particular, they both achieve an almost perfect result (only one false positive, no false negatives) using the mean+std threshold. The MPNet-PM model performs only slightly worse than the monolingual version, providing encouraging results in the perspective of future multilingual applications. The TF-IDF baseline performs worse than all the sentence embedding models.

Fig. 7 shows an example of matching using sentences from S1 to S19, which are the ones related to the argumentative nodes "Has celiac disease", "Has not celiac disease", "Is immunosuppressed", "Is not immunosuppressed". The matches are computed by the MPNet-S and the MPNet-P models using a threshold value of 0.65. The former achieves perfect precision but not perfect recall, and indeed we can see that it misses some matches, such as S8 and S10. The latter reaches perfect recall but not precision, which indicates the presence of false positives e.g. the pair S1 and S8.

To understand better how effectively the sentence have been modelled by the different embedding methods, we can use Principal Component Analysis to project the embeddings into a 2-dimensional space. Fig. 8 clearly shows that TF-IDF is the least effective at separating the sentences according to their nodes, while the MPNet methods are the most effective. It can also be seen that non-MPNet sentence embedding methods are not very effective in separating nodes that express opposite concepts (e.g., celiac vs non-celiac), while MPNet methods effectively separate sentences that express a negation (in green, blue, and cyan) from the others.

### 5.2. Test on the rest of the dataset

We test the best models and the associated best thresholds on the remaining 10 nodes and the related 38 natural language sentences. In particular, we use the three best fixed-value thresholds and the value mean+std previously obtained for each model. We do not compute a new mean+std value because our purpose is to validate the hyper-parameters selected in the previous step. The results are shown in Table 3, while Fig. 9 provides a graphical representation of the dataset sentences as they are encoded by the models.

The MPNet-S model achieves the best overall F1 score, while MPNet-PM is the one with the best recall among the cases where 100% precision is obtained. Both the models obtain their best F1 score for the lowest of the considered thresholds. Also, all three models obtain the lowest F1 with the higher threshold.

For the mean+std threshold, all the models have comparable F1 scores, MPNet-P is the least precise model, while the other models have

[13] All the implementations of the models are taken from http://www.sbert.net/.

perfect or almost perfect precision scores. The specific matches of obtained with each model are represented in Figs. 10, 11, and 12. By looking at them, we can see that the multilingual model completely fails to recognize two sentences, "I've never had bronchial asthma" and "I have never had an allergic reaction with latex". Regardng the MPNet-S model, its only false poitive is a match between the sentence "I went into anaphylactic shock before" and "I've never gone into anaphylactic shock before". This would prompt the argumentative model to ask for additional information on these topics, which surely is not ideal, but is not harmful either.

The experimental results are satisfactory and confirm the quality of our method. The fact that the multilingual model performs comparably to the monolingual models is encouraging in the perspective of future developments of multi-lingual chatbots that can be used not only by native speakers but also by tourists, migrants or refugees.

Some of the observed false positives might be particularly troubling in a real application since they mean that the system has misunderstood a sentence both for its real meaning but also for its negation, e.g. the sentence "I have a latex allergy" as "I do not have a latex allergy". The argumentative reasoning module would be able to easily detect such conflicts and in future works, we plan to include conflict resolution modules and procedures. A careful user experience design may also be able to mitigate the issue, for instance by displaying relevant pieces of information interactively as they are understood by the system.

We have never encountered cases where a sentence is misunderstood as something completely different, e.g. "I am celiac" as "I do not have any drug allergy", nor cases where a sentence is misunderstood only as its opposite. These cases are potentially harmful when the user has not provided other information regarding the latter aspect, so the argumentative model would not be able to detect the conflict. Such a problem could be addressed inside the language module, for example by establishing that a node is considered matched only if the user's input is similar to at least $K$ associated sentences, with $K$ being a new hyperparameter. Whether to enact such a strategy and the appropriate value for $K$ would depend on the specific use case and the number of sentences available in the KB for each node.

## 6. Conclusion

Dialogue systems are an increasingly popular class of AI systems that are nowadays used by many companies and institutions to provide services and information. The actual effectiveness of these systems is closely related to the trust that they inspire in the users. Usually, dialogue systems are designed to hide what happens behind the curtains, with the purpose to appear as "human" as possible and gaining the user's trust. Such a trend has not yet led to the desired outcome, and despite many efforts, users are still suspicious about dialogue systems, trusting them less than websites (Ischen et al., 2020). Also, while the design of the user experience is largely addressed in the literature (Rhim et al., 2022), surprisingly little technical work has been done yet in aspects such as the soundness of the answers, transparency of the computational process, and management of users' sensitive information.

The present work addresses important research questions: How to ensure data protection while providing personalized services to the individual? How to implement explainability in dialogue systems? How to implement a trustworthy dialogue system? It does so not in the abstract, but by presenting a concrete, modular architecture, and by evaluating its prototypical implementation on a simple but realistic case study. By doing so, it aims to bridge the gap between general ethical principles and practical realizations. Compare to mainstream dialogue systems architectures, our design is unusual, since it focuses primarily on data protection and transparency. It also addresses explainability, thanks to an Argumentation module that can justify the system's responses with KB facts that encode all the (sanitized) relevant information provided by the user. In this respect, a noteworthy feature of the Argumentation module is its ability to support reasoning over the conflicts between arguments, which lead to support or discard some responses. For example, our system can provide the user with justifications like '*Since you suffer from bronchial asthma, you can not get vaccinated at the vaccine site*'. We believe that justifying why a response cannot be given based on elements previously entered by the user is a good way to make the user understand the response and trust the system.

A Covid-19 vaccination case study was intended to illustrate how our proposal can be fit a real-world scenario, showing how users can interact with the system to retrieve information and obtain explanations about them. However, the architecture is of general applicability. Besides the case study, we assessed our system formally, with regard to the Argumentation module, and empirically, with regard to the Language module. Our empirical results, although in a small-sized case study, indicate that the concept is not only feasible but also, possibly, quite effective.

Future works include the design and implementation of strategies to resolve conflictual information in the argumentation module. Another possible extension could be giving the user an opportunity to directly correct matches. That would further improve transparency and reduce the number of false positives. However, that would also further complicate the interaction between users and the dialogue system, which may result in less trust. Also, our experimental results encourage us to explore the direction of a multi-lingual application. Finally, an evaluation conducted with human users would allow to evaluate the impact of these techniques in the perception of the users and whether they are perceived as trustworthy.

## Funding

This work was was partially supported by the EU H2020 ICT48 project "Humane AI Net" under contract #952026 and by the EU H2020 ICT49 project "StairwAI" under contract #101017142.

## CRediT authorship contribution statement

**Bettina Fazzinga:** Writing – original draft. **Andrea Galassi:** Writing – original draft. **Paolo Torroni:** Writing – original draft.

## Declaration of Competing Interest

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

## Appendix A. Sentences

The dataset is composed of 69 sentences related to 16 status nodes. The specific sentences are reported in Table A.1

**Table A.1**
Complete list of the sentences used in our case study and the argumentative node they are associated with.

| Node ID | Sent. ID | Sentence |
|---|---|---|
| N1 | S1 | I am celiac |
| N1 | S2 | I suffer from the celiac disease |
| N1 | S3 | I am afflicted with the celiac disease |
| N1 | S4 | I have the celiac disease |
| N1 | S5 | I recently found out to be celiac |
| N1 | S6 | I have suffered from celiac disease since birth |
| N2 | S7 | I do not have the celiac disease |
| N2 | S8 | I am not celiac |
| N2 | S9 | I do not suffer from the celiac disease |
| N2 | S10 | I am not afflicted with the celiac disease |
| N3 | S11 | I am not immunosuppressed |
| N3 | S12 | I do not suffer from immunosuppression |
| N3 | S13 | I am not afflicted with immunosuppression |

(*continued on next page*)

**Table A.1** (*continued*)

| Node ID | Sent. ID | Sentence |
|---|---|---|
| N4 | S14 | I am immunosuppressed |
| N4 | S15 | I suffer from immunosuppression |
| N4 | S16 | I am afflicted with immunosuppression |
| N4 | S17 | I do suffer from immunosuppression |
| N4 | S18 | I indeed suffer from immunosuppression |
| N4 | S19 | I recently found out to be immunosuppressed |
| N5 | S20 | I do not have any drug allergy |
| N5 | S21 | I do not suffer from drug allergies |
| N5 | S22 | I do not suffer from any drug allergy |
| N5 | S23 | I am not afflicted with any drug allergy |
| N5 | S24 | I do not have medication allergies |
| N5 | S25 | I do not have any medication allergy |
| N6 | S26 | I have a drug allergy |
| N6 | S27 | I do have a drug allergy |
| N6 | S28 | I have a serious drug allergy |
| N6 | S29 | I suffer from drug allergy |
| N6 | S30 | I am afflicted with drug allergies |
| N6 | S31 | I suffer from medication allergies |
| N7 | S32 | I do not suffer from bronchial asthma |
| N7 | S33 | I don't have bronchial asthma |
| N7 | S34 | I've never had bronchial asthma |
| N7 | S35 | I am not afflicted with bronchial asthma |
| N8 | S36 | I suffer from bronchial asthma |
| N8 | S37 | I have bronchial asthma |
| N8 | S38 | I am affected by bronchial asthma |
| N8 | S39 | I am afflicted with bronchial asthma |
| N9 | S40 | I suffer from diabetes |
| N9 | S41 | I am diabetic |
| N9 | S42 | I am affected by diabetes |
| N10 | S43 | I do not suffer from diabetes |
| N10 | S44 | I am not affected by diabetes |
| N10 | S45 | I am not diabetic |
| N10 | S46 | I don't have diabetes |
| N11 | S47 | I suffer from latex allergy |
| N11 | S48 | Im allergic to latex |
| N11 | S49 | I have a latex allergy |
| N11 | S50 | Latex causes me an allergic reaction |
| N12 | S51 | I do not suffer from latex allergy |
| N12 | S52 | Im not allergic to latex |
| N12 | S53 | I do not have a latex allergy |
| N12 | S54 | Latex does not cause me an allergic reaction |
| N12 | S55 | I have never had an allergic reaction with latex |
| N13 | S56 | I do not suffer from mastocytosis |
| N13 | S57 | I am not afflicted with mastocystosis |
| N13 | S58 | I do not have mastocystosis |
| N13 | S59 | Mastocystosis is not an health concern for me |
| N14 | S60 | I suffer from mastocytosis |
| N14 | S61 | I am afflicted with mastocystosis |
| N14 | S62 | I have a condition called mastocystosis |
| N15 | S63 | I have experienced a serious anaphylaxis in the past |
| N15 | S64 | I have had an anaphylactic reaction in the past |
| N15 | S65 | I have already had an anaphylactic reaction before |
| N15 | S66 | I went into anaphylactic shock before |
| N16 | S67 | Ive never experienced a serious anaphylaxis |
| N16 | S68 | Ive never had a serious anaphylactic reaction |
| N16 | S69 | Ive never gone into anaphylactic shock before |

## References


Altay, S., Hacquin, A.-S., Chevallier, C., & Mercier, H. (2021). Information delivered by a chatbot has a positive impact on COVID-19 vaccines attitudes and intentions. *Journal of Experimental Psychology: Applied.* https://doi.org/10.1037/xap0000400

Amiri, P., & Karahanna, E. (2022). Chatbot use cases in the COVID-19 public health response. *Journal of the American Medical Informatics Association, 29*(5), 1000–1010. https://doi.org/10.1093/jamia/ocac014

Barikeri, S., Lauscher, A., Vulic, I., & Glavas, G. (2021). RedditBias: A real-world resource for bias evaluation and debiasing of conversational language models, *vol. 1. ACL/IJCNLP* (pp. 1941–1955). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.151

Baroni, P., & Giacomin, M. (2009). Semantics of abstract argument systems. In G. R. Simari, & I. Rahwan (Eds.), *Argumentation in artificial intelligence* (pp. 25–44). Springer. https://doi.org/10.1007/978-0-387-98197-0_2.

Batet, M., & Sánchez, D. (2018). Semantic disclosure control: Semantics meets data privacy. *Online Information Review, 42*(3), 290–303. https://doi.org/10.1108/OIR-03-2017-0090

Boella, G., Gabbay, D. M., van der Torre, L. W. N., & Villata, S. (2010). Support in abstract argumentation. In P. Baroni, F. Cerutti, M. Giacomin, & G. R. Simari (Eds.), *Frontiers in artificial intelligence and applications: vol. 216. Computational models of argument: Proceedings of COMMA 2010, Desenzano del Garda, Italy, September 8–10, 2010* (pp. 111–122). IOS Press. https://doi.org/10.3233/978-1-60750-619-5-111.

Bray, J. R., & Curtis, J. T. (1957). An ordination of the upland forest communities of Southern Wisconsin. *Ecological Monographs, 27*(4), 325–349. https://doi.org/10.2307/1942268

Brixey, J., Hoegen, R., Lan, W., Rusow, J., Singla, K., Yin, X., Artstein, R., & Leuski, A. (2017). SHIHbot: A Facebook chatbot for sexual health information on HIV/AIDS. *SIGDIAL conference* (pp. 370–373). Saarbrücken, Germany: Association for Computational Linguistics. https://doi.org/10.18653/v1/W17-5544

Cayrol, C., & Lagasquie-Schiex, M. (2005). On the acceptability of arguments in bipolar argumentation frameworks, . In *Lecture notes in computer science: vol. 3571. ECSQARU* (pp. 378–389). Springer. https://doi.org/10.1007/11518655_33

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., & Specia, L. (2017). SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. *SemEval-2017* (pp. 1–14). Vancouver, Canada: Association for Computational Linguistics. https://doi.org/10.18653/v1/S17-2001

Chakaravarthy, V. T., Gupta, H., Roy, P., & Mohania, M. K. (2008). Efficient techniques for document sanitization. *CIKM '08: Proceedings of the 17th ACM conference on information and knowledge management* (pp. 843–852). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/1458082.1458194

Chalaguine, L. A., & Hunter, A. (2020). A persuasive chatbot using a crowd-sourced argument graph and concerns. In H. Prakken, S. Bistarelli, F. Santini, & C. Taticchi (Eds.), *Frontiers in artificial intelligence and applications: vol. 326. COMMA* (pp. 9–20). IOS Press. https://doi.org/10.3233/FAIA200487.

Chalaguine, L. A., & Hunter, A. (2021). Addressing popular concerns regarding COVID-19 vaccination with natural language argumentation dialogues. In J. Vejnarová, & N. Wilson (Eds.), *Lecture notes in computer science: vol. 12897. ECSQARU* (pp. 59–73). Springer. https://doi.org/10.1007/978-3-030-86772-0_5.

Charras, F., Dubuisson Duplessis, G., Letard, V., Ligozat, A.-L., & Rosset, S. (2016). Comparing system-response retrieval models for open-domain and casual conversational agent. *WOCHAT*. https://hal.archives-ouvertes.fr/hal-01782262.

Charwat, G., Dvorák, W., Gaggl, S. A., Wallner, J. P., & Woltran, S. (2015). Methods for solving reasoning problems in abstract argumentation - A survey. *Artificial Intelligence, 220*, 28–63. https://doi.org/10.1016/j.artint.2014.11.008

Chen, H., Liu, X., Yin, D., & Tang, J. (2017). A survey on dialogue systems: Recent advances and new frontiers. *SIGKDD Explor. Newsl., 19*(2), 25–35. https://doi.org/10.1145/3166054.3166058

Chesñevar, C. I., González, M. P., Maguitman, A. G., & Estevez, E. (2020). A first approach towards integrating computational argumentation in cognitive cities. In Y. Charalabidis, M. A. Cunha, & D. Sarantis (Eds.), *ICEGOV 2020: 13th international conference on theory and practice of electronic governance, Athens, Greece, 23–25 September, 2020* (pp. 25–32). ACM. https://doi.org/10.1145/3428502.3428506.

Cyras, K., Rago, A., Albini, E., Baroni, P., & Toni, F. (2021). Argumentative XAI: A survey. In Z. Zhou (Ed.), *Proceedings of the thirtieth international joint conference on artificial intelligence, IJCAI 2021, virtual event / Montreal, Canada, 19–27 August 2021* (pp. 4392–4399). ijcai.org. https://doi.org/10.24963/ijcai.2021/600.

Damerau, F. J. (1964). A technique for computer detection and correction of spelling errors. *Communications of the ACM, 7*(3), 171–176. https://doi.org/10.1145/363958.363994

Deriu, J., Rodrigo, Á., Otegi, A., Echegoyen, G., Rosset, S., Agirre, E., & Cieliebak, M. (2021). Survey on evaluation methods for dialogue systems. *Artificial Intelligence Review, 54*(1), 755–810. https://doi.org/10.1007/s10462-020-09866-x

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). BERT: pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, & T. Solorio (Eds.), *NAACL-HLT (1)* (pp. 4171–4186). Association for Computational Linguistics. https://doi.org/10.18653/v1/n19-1423.

Dinan, E., Fan, A., Williams, A., Urbanek, J., Kiela, D., & Weston, J. (2020). Queens are powerful too: Mitigating gender bias in dialogue generation. *EMNLP (1)* (pp. 8173–8188). Online: Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.656

Dos Santos Júnior, V. O., Castelo Branco, J. A., De Oliveira, M. A., Coelho Da Silva, T. L., Cruz, L. A., & Magalhes, R. P. (2021). A natural language understanding model COVID-19 based for chatbots. *2021 IEEE 21st international conference on bioinformatics and bioengineering (BIBE)* (pp. 1–7). https://doi.org/10.1109/BIBE52308.2021.9635248

Dung, P. M. (1995). On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence, 77*(2), 321–358. https://doi.org/10.1016/0004-3702(94)00041-X

Dung, P. M., Mancarella, P., & Toni, F. (2007). Computing ideal sceptical argumentation. *Artificial Intelligence, 171*(10–15), 642–674. https://doi.org/10.1016/j.artint.2007.05.003

Fazzinga, B., Flesca, S., & Furfaro, F. (2019). Complexity of fundamental problems in probabilistic abstract argumentation: Beyond independence. *Artificial Intelligence, 268*, 1–29. https://doi.org/10.1016/j.artint.2018.11.003

Fazzinga, B., Galassi, A., & Torroni, P. (2021a). An argumentative dialogue system for COVID-19 vaccine information. In P. Baroni, C. Benzmüller, & Y. N. Wáng (Eds.), *Lecture notes in computer science: vol. 13040. CLAR* (pp. 477–485). Springer. https://doi.org/10.1007/978-3-030-89391-0_27.

Fazzinga, B., Galassi, A., & Torroni, P. (2021b). A preliminary evaluation of a privacy-preserving dialogue system. In E. Cabrio, D. Croce, L. C. Passaro, & R. Sprugnoli (Eds.), *CEUR workshop proceedings: vol. 3015. Proceedings of the fifth workshop on natural language for artificial intelligence (NL4AI 2021) co-located with 20th international conference of the Italian association for artificial intelligence (AI*IA 2021), online event, November 29, 2021*. CEUR-WS.org. http://ceur-ws.org/Vol-3015/paper98.pdf

Galassi, A., Drazewski, K., Lippi, M., & Torroni, P. (2020). Cross-lingual annotation projection in legal texts. *COLING* (pp. 915–926). Barcelona, Spain (Online): International Committee on Computational Linguistics. https://doi.org/10.18653/v1/2020.coling-main.79

Galassi, A., Lippi, M., & Torroni, P. (2021). Attention in natural language processing. *IEEE Transactions on Neural Networks and Learning Systems, 32*(10), 4291–4308. https://doi.org/10.1109/TNNLS.2020.3019893

Gretz, S., Toledo, A., Friedman, R., Lahav, D., Weeks, R., Bar-Zeev, N., Sedoc, J., Sangha, P., Katz, Y., & Slonim, N. (2022). Benchmark data and evaluation framework for intent discovery around COVID-19 vaccine hesitancy. *CoRR*. arXiv preprint:2205.11966.

Hassan, F., Sánchez, D., Soria-Comas, J., & Domingo-Ferrer, J. (2019). Automatic anonymization of textual documents: Detecting sensitive information via word embeddings. *Trustcom/bigdatase* (pp. 358–365). IEEE. https://doi.org/10.1109/TrustCom/BigDataSE.2019.00055

Henderson, P., Sinha, K., Angelard-Gontier, N., Ke, N. R., Fried, G., Lowe, R., & Pineau, J. (2018). Ethical challenges in data-driven dialogue systems. *AIES* (pp. 123–129). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3278721.3278777

Hildebrandt, M. (2020). *Law for computer scientists and other folk*. Oxford University Press.

Humeau, S., Shuster, K., Lachaux, M., & Weston, J. (2020). Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. *ICLR*. OpenReview.net. https://openreview.net/forum?id=SkxgnnNFvH

Ischen, C., Araujo, T., Voorveld, H., van Noort, G., & Smit, E. (2020). Privacy concerns in chatbot interactions. In A. Følstad, T. Araujo, S. Papadopoulos, E. L.-C. Law, O.-C. Granmo, E. Luger, & P. B. Brandtzaeg (Eds.), *Chatbot research and design* (pp. 34–48). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-39540-7_3.

Iwendi, C., Moqurrab, S. A., Anjum, A., Khan, S., Mohan, S., & Srivastava, G. (2020). N-sanitization: A semantic privacy-preserving framework for unstructured medical datasets. *Computer Communications, 161*, 160–171. https://doi.org/10.1016/j.comcom.2020.07.032

Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., Wang, F., & Liu, Q. (2020). TinyBERT: Distilling BERT for natural language understanding. *Findings of the association for computational linguistics: Emnlp 2020* (pp. 4163–4174). Online: Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.findings-emnlp.372

Judson, T. J., Odisho, A. Y., Young, J. J., Bigazzi, O., Steuer, D., Gonzales, R., & 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*(9), 1450–1455. https://doi.org/10.1093/jamia/ocaa130

Karami, M., Mosallanezhad, A., Mancenido, M. V., & Liu, H. (2021). "Let's eat grandma": When punctuation matters in sentence representation for sentiment analysis. *CoRR*. abs/2101.03029.

Kenter, T., & de Rijke, M. (2015). Short text similarity with word embeddings, . In *CIKM '15CIKM* (pp. 1411–1420). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/2806416.2806475

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A. P., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., Toutanova, K., Jones, L., Kelcey, M., Chang, M., Dai, A. M., Uszkoreit, J., Le, Q., & Petrov, S. (2019). Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics, 7*, 452–466. https://doi.org/10.1162/tacl_a_00276

Le, Q. V., & Mikolov, T. (2014). Distributed representations of sentences and documents, . In *JMLR workshop and conference proceedings: vol. 32. ICML* (pp. 1188–1196). JMLR.org.http://proceedings.mlr.press/v32/le14.html

Li, B., Vorobeychik, Y., Li, M., & Malin, B. A. (2017). Scalable iterative classification for sanitizing large-scale datasets. *IEEE Transactions on Knowledge and Data Engineering, 29*(3), 698–711. https://doi.org/10.1109/TKDE.2016.2628180

Lin, C., Bethard, S., Dligach, D., Sadeque, F., Savova, G., & Miller, T. A. (2020). Does BERT need domain adaptation for clinical negation detection? *Journal of the American Medical Informatics Association, 27*(4), 584–591. https://doi.org/10.1093/jamia/ocaa001

Lison, P., Pilán, I., Sánchez, D., Batet, M., & Øvrelid, L. (2021). Anonymisation models for text data: State of the art, challenges and future directions. *ACL/IJCNLP (1)* (pp. 4188–4203). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.323

Liu, H., Dacon, J., Fan, W., Liu, H., Liu, Z., & Tang, J. (2020). Does gender matter? Towards fairness in dialogue systems. *COLING* (pp. 4403–4416). Barcelona, Spain (Online): International Committee on Computational Linguistics. https://doi.org/10.18653/v1/2020.coling-main.390

Luo, L., Huang, W., Zeng, Q., Nie, Z., & Sun, X. (2019). Learning personalized end-to-end goal-oriented dialog. *Proceedings of the AAAI Conference on Artificial Intelligence, 33*(01), 6794–6801. https://doi.org/10.1609/aaai.v33i01.33016794

Miner, A. S., Laranjo, L., & Kocaballi, A. B. (2020). Chatbots in the fight against the COVID-19 pandemic. *npj Digital Medicine, 3*(1). https://doi.org/10.1038/s41746-020-0280-0

Modgil, S., Toni, F., Bex, F., Bratko, I., Chesñevar, C. I., Dvořák, W., Falappa, M. A., Fan, X., Gaggl, S. A., García, A. J., González, M. P., Gordon, T. F., Leite, J., Možina, M., Reed, C., Simari, G. R., Szeider, S., Torroni, P., & Woltran, S. (2013). *The added value of argumentation*. In S. Ossowski (Ed.) (pp. 357–403)). Dordrecht: Springer Netherlands.

Mohamad Suhaili, S., Salim, N., & Jambli, M. N. (2021). Service chatbots: A systematic review. *Expert Systems with Applications, 184*, 115461. https://doi.org/10.1016/j.eswa.2021.115461

Nguyen, H., & Cavallari, S. (2020). Neural multi-task text normalization and sanitization with pointer-generator. *Proceedings of the first workshop on natural language interfaces* (pp. 37–47). Online: Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.nli-1.5

Pennington, J., Socher, R., & Manning, C. (2014). GloVe: Global vectors for word representation. *EMNLP* (pp. 1532–1543). Doha, Qatar: Association for Computational Linguistics. https://doi.org/10.3115/v1/D14-1162

Pilán, I., Lison, P., Øvrelid, L., Papadopoulou, A., Sánchez, D., & Batet, M. (2022). The text anonymization benchmark (TAB): A dedicated corpus and evaluation framework for text anonymization. *CoRR*. arXiv preprint:2202.00443.

Queudot, M., Charton, E., & Meurs, M.-J. (2020). Improving access to justice with legal chatbots. *Stats, 3*(3), 356–375. https://doi.org/10.3390/stats3030023

Rach, N., Langhammer, S., Minker, W., & Ultes, S. (2018). Utilizing argument mining techniques for argumentative dialogue systems. In L. F. D'Haro, R. E. Banchs, & H. Li (Eds.), *Lecture notes in electrical engineering: vol. 579. IWSDS* (pp. 131–142). Springer. https://doi.org/10.1007/978-981-13-9443-0_12.

Rajendran, J., Ganhotra, J., Singh, S., & Polymenakos, L. (2018). Learning end-to-end goal-oriented dialog with multiple answers. *EMNLP* (pp. 3834–3843). Brussels, Belgium: Association for Computational Linguistics. https://doi.org/10.18653/v1/D18-1418

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. *EMNLP/IJCNLP (1)* (pp. 3982–3992). Hong Kong, China: Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1410

Reimers, N., & Gurevych, I. (2020). Making monolingual sentence embeddings multilingual using knowledge distillation. In B. Webber, T. Cohn, Y. He, & Y. Liu (Eds.), *EMNLP* (pp. 4512–4525). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.365.

Rhim, J., Kwak, M., Gong, Y., & Gweon, G. (2022). Application of humanization to survey chatbots: Change in chatbot perception, interaction experience, and survey data quality. *Computers in Human Behavior, 126*, 107034. https://doi.org/10.1016/j.chb.2021.107034

Rosenfeld, A., & Kraus, S. (2016). Strategical argumentative agent for human persuasion. *ECAI'16: ECAI* (pp. 320–328). NLD: IOS Press. https://doi.org/10.3233/978-1-61499-672-9-320

Saglam, R. B., Nurse, J. R. C., & Hodges, D. (2021). Privacy concerns in chatbot interactions: When to trust and when to worry. In C. Stephanidis, M. Antona, & S. Ntoa (Eds.), *Communications in computer and information science: vol. 1420. HCII* (pp. 391–399). Springer. https://doi.org/10.1007/978-3-030-78642-7_53.

Sánchez, D., Batet, M., & Viejo, A. (2014). Utility-preserving privacy protection of textual healthcare documents. *Journal of Biomedical Informatics, 52*, 189–198. https://doi.org/10.1016/j.jbi.2014.06.008

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *The 5th workshop on energy efficient machine learning and cognitive computing @ neurips*. arXiv preprint:1910.01108.

Schubel, L. C., Wesley, D. B., Booker, E., Lock, J., & Ratwani, R. M. (2021). Population subgroup differences in the use of a COVID-19 chatbot. *NPJ Digit. Med., 4*(1), 30. https://doi.org/10.1038/s41746-021-00405-8

Song, K., ando Tao Qin, X. T., Lu, J., & Liu, T. (2020). Mpnet: Masked and permuted pre-training for language understanding. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, & H. Lin (Eds.), *Neurips*.https://proceedings.neurips.cc/paper/2020/hash/c3a690be93aa602ee2dc0ccab5b7b67e-Abstract.html

Szarvas, G., Farkas, R., & Busa-Fekete, R. (2007). Research paper: State-of-the-art anonymization of medical records using an iterative machine learning framework. *Journal of the American Medical Informatics Association, 14*(5), 574–580. https://doi.org/10.1197/jamia.M2441

Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., & Zhou, M. (2020). Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, & H. Lin (Eds.), *Neurips*.https://proceedings.neurips.cc/paper/2020/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

Wen, T.-H., Vandyke, D., Mrkšić, N., Gašić, M., Rojas-Barahona, L. M., Su, P.-H., Ultes, S., & Young, S. (2017). A network-based end-to-end trainable task-oriented dialogue system. *EACL (1)* (pp. 438–449). Valencia, Spain: Association for Computational Linguistics.https://www.aclweb.org/anthology/E17-1042

Xu, L., Zhou, Q., Gong, K., Liang, X., Tang, J., & Lin, L. (2019). End-to-end knowledge-routed relational dialogue system for automatic diagnosis. *AAAI* (pp. 7346–7353). AAAI Press. https://doi.org/10.1609/aaai.v33i01.33017346

Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018). Recent trends in deep learning based natural language processing [review article]. *IEEE Computational Intelligence Magazine, 13*(3), 55–75. https://doi.org/10.1109/MCI.2018.2840738

Zhao, T., & Eskénazi, M. (2016). Towards end-to-end learning for dialog state tracking and management using deep reinforcement learning. *SIGDIAL conference* (pp. 1–10). The Association for Computer Linguistics. https://doi.org/10.18653/v1/w16-3601

Zhu, X., Li, T., & de Melo, G. (2018). Exploring semantic properties of sentence embeddings. *ACL (2)* (pp. 632–637). Association for Computational Linguistics. https://doi.org/10.18653/v1/P18-2100