Chatbot Theory*

A Naïve and Elementary Theory for Dialogue Management

Francisco S. Marcondes[1] [0000−0002−2221−2261],
José João Almeida[1] [0000−0002−0722−2031], and
Paulo Novais[1] [0000−0002−3549−0754]

University of Minho, Algoritmi Centre, ISLab
id7515@alunos.uminho.pt and {jj|pjon}@di.uminho.pt
http://algoritmi.uminho.pt/

Abstract. Due to the increasing interested and use of chatbot, its properties and operation possibilities shall be proper realized matching both safety and security issues as well as present the several uses and compositions that this technology supports. This paper focus is on dialogue management since it is considered the core of a chatbot. The dialogue manager is responsible to, more than to transform an input sentence into an output one, hold the illusion of a human conversation. In this sense, it is presented an inceptive theoretical framework through a formal way for chatbots that can be used as a reference to explore, compose, build and discuss chatbots. The discussion is performed mostly on ELIZA since, due to its historical records, it can be considered an important reference chatbot, nevertheless, the proposed theory is compatible with the most recent technologies such those using machine and deep learning. The paper then presents some sketchy instances in order to explore the support provided by the theory.

Keywords: chatbot · formal theory · dialogue manager · mechanical dialogue

1 Introduction

There is a growing interest in chatbots (see figure 1a) as it uses in the internet increases (it may reach 85% of chat interactions in 2020 [10]). These applications are being used through a wide spectrum and, as an outcome, raising issues as when SIMSIMI starts to threat kidnapping [6].

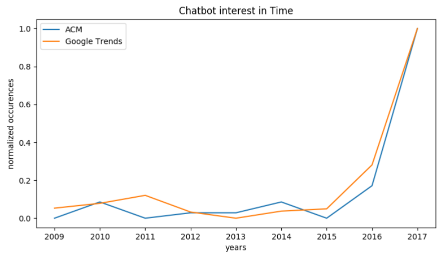
To allow the use of chatbots in increasingly diverse contexts and also to handle the concerns that arise from this, a theory is needed [13].

This paper intention is then to present an inceptive chatbot theory as proof of concept on how a theoretical view may help on matters as such. A theory as such provides an underlying framework that can be used when examining mechanical dialogue phenomena and explore design and behaviour decisions when building a chatbot application. Paper Objective

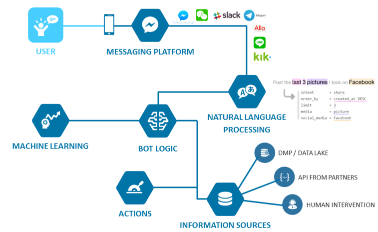
1.1 A Brief Chatbot Review

A chatbot is a device intended to perform *social interactions* like a human would do [17, 18, 6]. This is the cornerstone for chatbots, otherwise, it may easily become a question and answer, Dialogue Illusion

* This paper has been supported by COMPETE: POCI-01-0145-FEDER-0070 43 and FCT – Fundação para a Ciência e Tecnologia - Project UID/CEC/ 00319/2013.



a)



b)

Fig. 1. a) The chart shows the variation of chatbot interest in time by ACM and Google Trends. The search keyword used on both was *chatbot* and the values were normalized by Min-Max scaling to provide a better comparison. b) A chatbot reference architecture [7].

information retrieval, or another system as such. In other words, to be a chatbot an application must create the illusion of an actual dialogue between people [6].

Mechanical Dialogue

Every chatbot as a computer program is reducible to a Turing Machine which implies that it may only perform a mechanical behaviour [17, 1]. Within the dialogue domain, this results in a *mechanical dialogue* conception as a narrower version of scripted dialogues as those used for telemarketing. The art is how to hold the illusion of human conversation through mechanical procedures.

Dialogue Manager Conception

The responsibility to create and sustain the illusion of social interaction is of the *dialogue manager*. Figure 1b present a reference stack for chatbot technology aiding the understanding of how the different components interact; the dialogue manager is exhibited as “Bot Logic”. Then to fulfil its responsibility the dialogue manager articulates all the other components to generate a “proper” output [8]. The word “proper” means an output that nurtures the illusion of a conversation.

Dialogue Manager Operation

Roughly, the behaviour of the dialogue manager can be described as a mapping function $f : I \rightarrow O$ where I stands for input and O for output; both being sets of sentences. The mapping is used to be performed by a determinitic or statical procedures. Roughly, on the deterministic approach, all possible mappings are defined *a priori* by a set o rules resulting in a behavior as $if\ i \in I\ then\ o \in O$. On the statistical approach, first, a dataset is turned into a dictionary of words and probability relations among them; when an input is received is then determined the probability of the next word as $f(f(f(f(i) : o_1) : o_2) : \ldots) : o_n)$. Deterministic approaches perform a straightforward mapping while statistical approaches a probabilistic one. On the edge, where an input-sentence matches perfect proximity (equals ‘1.0’), it tends to work as a deterministic chatbot.

Formal Theory

A mathematical theory is a set of rules that determine the production of mathematical models, *e.g.* the number theory [9]. In this sense, the set of rules that settle a chatbot model is its theory and a chatbot is an instance of that model. Even intuitionism being the suitable mathematical paradigm to programming [16], as it is not yet possible to define each element *a priori* and also avoid supposition, generality and infinity it suits better develop the theory through formalism [3] tending to intuitionism as it nears to become an instance as a computer program.

2 The Dialogue Manager Theory

Elementary and Naïve Theory

As this paper intends to present an inceptive mathematical theory, it is kept as simple as possible. The proposed calculus is elementary (does not support variables) and the presentation is naïve (the focus holds on the probe and not in full formalization). Let’s then proceed with the presentation.

The smallest unit of expression in English is a phrase (or an utterance on speaking). A phrase requires neither subject or predicate as long it express something, *e.g.* the interjection "woah!" is a phrase. Therefore a dialogue is a sequence of phrases and phrases assemblages among actors. During a dialogue, each actor says a phrase that answers to a previous one, start a new conversation topic, or both. Within a dialogue, there are also relevant non-verbal markers as the silence that can be associated with many semantics given a context. Chat Phrase Set

The proposed formalization is composed of a set of phrases name P (the definition of the phrase conception is witining postponed to each dialogue manager design). Linked to P there is a Φ set of functions over P as $\{\varphi : P \to P | \varphi \in \Phi\}$. The difference between two functions φ and φ' in Φ is given by the mapping rule bounded to each one. The aimed formalism requires another set to hold the ending phrases denoting the end of a conversation, this is called the F set. These lead to the formulation stated in definition 1.

Definition 1 (Dialogue Manager). *A* dialogue manager *is an extensible 3-tuple* $\{P, \Phi, F\}$ *where,*

P *is the set of chat-phrases plus* ξ *as silence mark.*
Φ *is the set of attached functions*[1] *over a set as* $\varphi \bowtie P$ *and of the commonly used set operators.*
$F \subseteq P$ *is a set of chat-phrases expressing the end of dialogue (followed by* $\odot$*).*

The symbol $\cdot$ *denotes the dialogue start and symbol* $\odot$ *the dialogue end.*

Then, given a dialogue manager $B = \{P, \Phi, F\}$ a *mechanical dialogue* is a sequence Mechanical Dialogue Def.

$$\cdot p \in P \overset{(\varphi \bowtie P)\in\Phi}{\vdash} p' \in P \overset{(\varphi' \bowtie P)\in\Phi}{\vdash} \ldots \overset{(\varphi' \bowtie P)\in\Phi}{\vdash} p'' \in P \overset{(\varphi'' \bowtie F)\in\Phi}{\vdash} f \in F\odot$$

Whenever the chosen paradigm, the dialogue manager definition turns out to be something like $\cdot Hi \vdash Hello \vdash \ldots \vdash Bye \vdash See'ya\odot$ (refer to example 1 for a full presentation). This example suites the mechanical dialogue definition as φ relates to a dialogue start protocol that then turns in to φ' as "common dialogue" and then φ'' relates to a dialogue end protocol over the F set. Definition 1 is said to be extensible as it supports the construction of other subsets of P than just F. It would be a S set to handle the dialogue start protocol which would result in the function $(\varphi \bowtie S) \in \Phi$ within the mechanical dialogue definition. Definition Extensions

Since $B = \{P, \Phi, F\}$ and $F \subseteq P$; therefore $B = \{P, \Phi\}$ (the F set is required to trace the end of the dialogue). So, given a set-builder in the form $X = \{p \in P | \Xi(p)\}$, that results into a subset X with elements on P that $\Xi(p)$ is true; then $F = \{p \in P | p \text{ is a finishing phrase}\}$. Any other subset can be defined as such. Each subset must have a function bound to it, nevertheless, a function as $\varphi \bowtie F \subset P$ is also a function $\varphi \bowtie P$. This is used to trace special conditions and perform separation of concerns as intended by [12, 5]. Dialogue Manager Builder

A function implies the existence of a Cartesian product on P, *i.e.* $(\varphi : P \to P) = P \times P$ producing an implicit Δ set composed by binary relations in the form $(p_a, p_b) \in \Delta$. Suppose $P = \{p_a, p_b, p_c\}$, it requires the existence of an implicit $P \times P$ set called Δ to a function operates on it. Let $\Delta = [(p_a[p_b, p_c]), (p_b[p_a, p_c]), (p_c[p_a, p_b])]$, if $\delta = (p_a[p_b, p_c]) \in \Delta$ is called, shall it result in p_b, p_c or $[p_b, p_c]$? An answer to it depends on the chatbot model under design and therefore such is also kept wittingly undefined avoiding to fast to a particular approach.

In addition to the functions, Φ set is also formed by the commonly used set operators. Suppose a dialogue manager A and a dialogue manager B. If they are disjoint $(A \neq B)$, *i.e.* if Dialogue Manager Operations

[1] In this paper a function $f(a) : b$ is denoted by $a \vdash b$.

$A \cap B = \emptyset$ then the formation of a joint dialogue manager is quite straightforward as $K_{A \cup B} = \{P_{(P' \in A) \cup (P'' \in B)}, \Phi_{(\Phi' \bowtie A) \cup (\Phi'' \bowtie B)}, F\}$, it however may need some refactoring on the F set and attached functions. If there are overlapping data or rules a more profound refactoring may be needed, anyway as the whole proposal is formal a tool may aid the conflict resolution. Otherwise, if $P' \in A = P'' \in B$ $(\forall x((x \in P') \leftrightarrow (x \in P'')))$, the two dialogue managers can be considered similar $(A \equiv B)$. If the same happens on Φ as $\forall \varphi((\varphi' \bowtie P' \in A) = (\varphi'' \bowtie P'' \in B))$ then A and B are the same chatbot.

Dialogue Turns Natural deduction structure is a suitable way to handle dialogue "turns" in the form (strict) $\dfrac{\text{Premisses}}{\text{Output-Phrase}}\,\text{Input-Phrase}$, for instance $\dfrac{P \qquad \varphi \bowtie P \qquad \Gamma}{\text{b}}\,\text{a}$ (Γ denotes the preceding turns). Since it is possible to provide more than one function to P, the function itself is a premisse. It can be also in the form (rough) $\dfrac{\text{Input-Phrase} \in P}{\text{Output-Phrase} \in P}\,\varphi$. See example 1.

Example 1 (Dialogue Manager).

Let $B = \{P, \Phi, F, S\}$

P is a set of any phrases
$\Phi = \{\varphi \bowtie S, \varphi' \bowtie P, \varphi'' \bowtie F, \ldots\}$
$F = \{Bye, See'ya, \ldots\}$
$S = \{Hi, Hello, \ldots\}$

$$\dfrac{S = \{Hi, Hello, \ldots\} \qquad \varphi \bowtie S \qquad \cdot}{\text{Hello}}\,\text{Hi}$$
$$\ddots$$
$$\dfrac{P \qquad \varphi' \bowtie P \qquad \Gamma}{\text{y}}\,\text{x}$$
$$\ddots$$
$$\dfrac{F = \{Bye, See'ya, \ldots\} \qquad \varphi'' \bowtie F \qquad \Gamma}{\text{See'ya}}\,\text{Bye}$$

$\odot$

All this is valid both to deterministic and statistical approaches. Within the deterministic approach, the use of the presented formalism is straightforward. In the statistical approaches, the sets can be established by annotations in the data set; the Φ set is defined by the statistical method being used. The use of several functions within the dialogue manager definition is an illustration, however, most of the current chatbots use just one function.

3 Dialogue Manager Framework

In addition to the theory, the composition of a dialogue manager requires five concerns to be handled, as presented in table 3, ranging from operation to behaviour related issues (an operation is a behaviour declaration and behaviour is an operation realization [14]).

Concern	Instances	Affair
Paradigm	Deterministic, Statistical, *etc.*	Operation
Realization	Boolean, Fuzzy, Markov, Deep Learning *etc.*	↓
Strategy	Direct Map, SWM, Seq2Seq, *etc.*	
Approach	Phrase-Reply, Pattern-Template, Frame-slot, *etc.*	↑
Rule-set	Rogerian, Eckmannian, queue, *etc.*	Behaviour

Table 1. Dialogue Manager Framework.

ELIZA, for instance, is a deterministic (paradigm) boolean (realization) chatbot modelled as an SWM (strategy) pattern-template (approach) with input priority defined by Rogerian psychology (rule-set) and output priority by a randomized queue (rule-set). Therefore, once the chatbot concerns are classified, it is then possible to think about variations and compositions to be performed.

The proposed theory shall then provide an underlying support for the treatment and discussions of all those concerns. In other words, a full chatbot must address all those concerns grounded in a theory that supports it. In this sense, the Dialogue Manager Framework is the theory as presented in definition 1 and the concerns as exhibited in 3.

3.1 Boolean Dialogue Manager Realization Instance

The Boolean operators *cf.* [15] can be described as functions on Φ and stated as in definition 2 being an example of deterministic dialogue manager.

Definition 2 (Boolean Dialogue Manager).
A boolean dialogue manager is 3-tuple $\{P, \Phi, F\}$ *as in definition 1 and*

$p \vee p'$ *is an expectation of* p *or* p' *due to an input.*
$p \wedge p'$ *is a sentence composed by the phrases* p *and* p'.
$p \vdash p'$ *is a function* f *that produces* p' *from* p.
$\neg p$ *mean that there is no* p *which is meaningless the* $\neg P \vdash \perp$
$\perp$ *is an absurd statement within a chat context that breaks the illusion of dialogue.*

The implicative operator can be used in the same sense of specific defined functions in Φ and results as well in a sequence on P as $p \vdash p' \vdash \ldots \vdash p''$. The alternative operator can be used when choosing between phrases as $\dfrac{((p \vdash p') \vee (\xi \vdash p''))}{p'}\, p$ or $\dfrac{((p \vdash p') \vee (\xi \vdash p''))}{p''}\, \xi$. The conjunctive operator can be used to handle sentences composition and splitting of sentences to handle information antecipation and dialogue initiative turns $\dfrac{P \qquad \varphi \qquad \Gamma}{(p'' \wedge p''')}\, (p \wedge p')$. However, the negative operator may result into a nonsense output as presented in the lemma 1 and as a weak claim, negation shall be avoided within boolean dialogue managers since it leads to a [pragmatic] contradiction.

Boolean Operators

Lemma 1 (The negation chat-implication leads to an absurd). *Let* $P = \{p_a, p_b, \ldots, p_n\}$
Since $\dfrac{p_a \vdash p_b}{p_b}$, *Thus* $\dfrac{p_a \nvdash p_b}{\bar{P}}$, *Then* $S = \{p_a, \ldots, p_n\}$ *being an absurd chat-sentence to output.*
$\therefore (p_a \nvdash p_b) \rightarrow \perp$

It shall be highlighted that these operators refer to the logical relations between phrases and not to their semantics. Let $p =$ 'the snow is white' then $p \wedge p'$ is $TRUE$ if both are $\in P$, are called, *etc.* As well $\neg p$ does not means that 'the snow is not white' but that there is no such statement.

3.2 Markovian Dialogue Manager Realization Instance

The Markov chains *cf.* [15] can be described as functions on Φ and stated as in definition 3 being an example of statistical dialogue manager.

Definition 3 (Markovian Dialogue Manager).
A markovian dialogue manager is 3-tuple $\{P, \Phi, F\}$ *as in definition 1 where*

Φ *includes the probability function as* $\varphi_\delta(P_{t+1} = p | P_t = p_t)$

Δ is the implicit set that holds the probabilities of $\varphi_\delta \bowtie P$

Markov Example

According to the Markov Chain formalism, the function denoted by φ_δ is a transition probability function and is used to calculate the phrase p in the time $t+1$ given a sequence of p_t phrases. As an example, let $\Delta = \begin{array}{c} \\ Hi \\ Hello \\ Bye \\ See'ya \end{array} \begin{array}{c} \begin{array}{ccccc} Hi & Hello & Bye & See'ya & \ldots \end{array} \\ \begin{pmatrix} 0.27 & 0.29 & 0.12 & 0.10 & \epsilon \\ 0.28 & 0.26 & 0.14 & 0.12 & \epsilon \\ 0.13 & 0.11 & 0.26 & 0.28 & \epsilon \\ 0.10 & 0.11 & 0.29 & 0.25 & \epsilon \end{pmatrix} \end{array}$ then $\varphi_\delta(P_1 = Hello|P_0 = Hi)$ and $\varphi_\delta(P_{t+n+1} = Bye|P_{t+n} = \ldots)$, $\varphi_\delta(P_{t+n+2} = See'ya|P_{t+n+1} = Bye)$. As Bye and $See'ya$ and in the set F, it can be attached the end symbol $\odot$ in the time P_{t+n+2}.

3.3 ELIZA and MARK

Historical Chatbots and Current Tech.

Two historical representative chatbots are ELIZA and Mark v Shaney (MARK). They are said to be representative since ELIZA inspired approaches are currently being used in top of hedge chatbots as MITSUKU, the 2017 Loebner Prize winner. MARK inspired approaches are also currently being used as by the seq2seq machine learning chatbots.

ELIZA is a deterministic chatbot built to simulate a Rogerian psychotherapist [18] which was also considered to be actually used as a first line of psychological support [2]. MARK is a statistical chatbot built to discuss in the Usenet newsgroups [4], it took other users positions and through Markov Chains generates a reply that was able to fool several users.

Behaviour Description

Roughly, the behaviour of ELIZA is based on SWM (Split-Weight-Map), consisting in split an input-sentence into phrases according to punctuation marks. Match them to a pattern weighting them and by polling the phrase with the highest weight raises its template.

Also roughly, The behaviour of MARK is given a data set, it calculates the sequence probability of each word (phrase here is realized as word). Then, choosing the first word as the time P_0 it creates a sequence $p_0, \ldots, p_n$ to time P_n (each "time" is a word).

3.4 Dialogue Manager Sketches

The Framework Rationale

It can be realized that ELIZA uses the SWM strategy with the pattern-template approach and Rogerian rule-set. These, however, are design decisions of ELIZA bot. There are several variations that can be performed generating several chatbots.

It is possible as an instance to change ELIZA's approach from pattern-template to phrase-reply yielding an elementary chatbot. It is also possible to change the deterministic mapping procedure of SWM to a probabilistic one as of MARK. As well it is possible to build variations on MARK to handle phrases (not words) or perform weighting as ELIZA does to guide the probabilistic choice.

This suggests that allied to the proposed theory, the framework helps in the exploration of dialogue management conceptions. To a further inception it is present some sketches to be considered. To an easier organization of ideas, the discussion will be focused on ELIZA-like dialogue managers.

Dialogue Protocols

There is an aforementioned statement for "dialogue protocols" to handle special circumstances such as the dialogue start, end, *etc.* A deterministic way to perform such scripts is by a state machine based approach as, let $\varphi \bowtie S$ produce the implicit $\Delta' = \begin{array}{c} \\ s_0 \\ s_1 \\ s_2 \end{array} \begin{array}{c} \begin{array}{cccc} s_0 & s_1 & s_2 & p \in P \end{array} \\ \left(\begin{array}{ccc|c} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right) \end{array}$ that acts as an incidence matrix, and then resulting in the chat sequence $\Gamma \vdash s_0 \vdash s_1 \vdash s_2 \vdash p \vdash \Gamma$.

Emotion Weighting

In addition, the ELIZA Rogerian rule-set can be replaced by an Eckmanninan one. Roughly, Eckmann's model of basic emotions suggests that human emotions range between anger, disgust, fear, happiness, sadness, and surprise as positive or negative; it is then possible to classify an emotional value of some keywords creating a word-emotion lexicon as the one proposed in [11]. For an instance, Rogerian-based set can be something like $P_r = \{$('sorry', 0), ('remember', 5), ('dream', 3), ('my', 3), ('name', 15), $\ldots\}$ becoming $P_e = \{$('sorry', -1), ('remember', neutral), ('dream', neutral), ('my', -1), ('name', neutral), $\ldots\}$.

Blended Weighting

In addition to substitute a particular strategy it is also possible to perform a mixed approach. Let P_r be a Rogerian set and P_e an Eckmann set, and a resulting $P = P_r \cup P_e$ becomes $P_{r \cup e} = \{$('sorry', (0, -1)), ('remember', (5, neutral)), ('dream', (3, neutral)), ('my', (3, -1)), ('name', (15, neutral)), $\ldots\}$, assuming that P is accordingly fashioned, it is possible to perfom a $\vdash_{r \wedge e}$ function returning $P_r =$ (('Is it yours?', 1), ('Why bother saying that is yours?', 0)), $P_{e_1} =$ (neutral, [('Is it yours?', 1), ('Why bother saying that is yours?', 0)]) and P_{e_2} =(neutral, [('My name is Bot', 1)]). Then $(P_{e_1} \cup P_{e_2}) \cap P_r$ may be the result P set.

Composed Input Weighting

It is then possible to perform a mixed strategy as $P = P_r \cup P_e$ becoming $P_{r \cup e} = \{$('sorry', (0, -1)), ('remember', (5, neutral)), ('dream', (3, neutral)), ('my', (3, -1)), ('name', (15, neutral)), $\ldots\}$ providing a $\vdash_{r \wedge e}$ implication. It is also possible to define alternative strategy in the form of $\vdash_{r \vee e}$. This means that to each interaction the bot shall respond according to Rogerian or Eckmannian.

Safety Awareness

A possible application of such in addition to the dialogue protocols is to perform safety awareness monitoring *fear* if a fear phrase is detected then Eckmann strategy arises otherwise Rogerian strategy is used. Then, being $Z \subset P$ a set of fear keywords as $Z =$ (afraid, affliction, adverse, $\ldots$) as $if\ \exists z_i$ in p_j *then* z_i script start. Such script may lead to a decision tree modelled as a *dialogue protocol* to establish the possibility of risk of hazard. An example for fire awareness dialogue is presented in example 2. Note that a *ad hoc* dialogue *pragmatics* is being performed.

Example 2 (Fire Alert Chat Example). Let Z a set of fear keywords as Z= (afraid, affliction, $\ldots$)

$$\dfrac{\dfrac{\dfrac{\Gamma \vdash_r \Gamma}{\text{I'm afraid about my oven} \vdash_e \text{Why do you say so?}}}{\ddots}}{\dfrac{}{\Gamma\text{, Fire is starting...} \vdash_e \text{Calling the fire brigade.}}}$$

$\ddots$ This may continue with fire-fight specific advices

4 Results and Discussion

This paper starts from the assumption that the dialogue manager is the component responsible for conducting the mechanical dialogue in order to provide a proper social interaction by creating and sustaining the illusion of an actual human conversation. In this sense, the dialogue manager is considered the core of the chatbots that articulates the other components to accomplish its task (refer to figure 1b). It then presents a partially defined yet sufficient dialogue management theory.

The proposal is helpful to handle the scientific unsetting provoked by the several terminologies and behaviour used in different chatbot IDEs. Reducing each IDE to a theory as proposed in this paper leads to a common ground allowing discussions and comparisons among them.

Also, as it is based on formal paradigm this paper theory allows a more flexible exploration of chatbot concepts and issues than those that would be performable through IDEs. It likewise provides the choice and synthesis possibility of chatbots but keeping the relationship with computation.

Another outcome of this paper for chatbots is the possibility of a development process. Artificial intelligence systems commonly lack the support of software processes when using "usual" approaches; roughly the specification turns to be quite simple and most of the complexity raises during implementation. The use of the proposed theory as a chatbot framework allow a proper setting for separation of concerns and project life cycles.

Mechanical dialogues and dialogue management are related subjects since the former performs a realization of the prior. As an application building, they refer to different concerns; during analysis what is being analysed is the intended or convenient *mechanical dialogue* to be realized by the *dialogue manager* during design. As a research theme, the first is of interest mainly by linguistics and cognitive scientists while the second being primarily technological artefact concerns to informatics related people. Nevertheless, the presented theory is able to provide support to both of them.

As related works comparison, performing a brief survey on the theme, the key-word "chatbot" in ACM digital library returns 110 entries (the dispersion in time is depicted in figure 1a). The 10% of the most relevant papers (according to ACM criteria) are centred in two main interests: dialogue improvements, development process and frameworks (6 papers); and applications, human-computer interfaces and post-human concerns (5 papers). The presented theory fits in neither of those groups as it is a theory and both groups are mainly concerned with implementations, in this sense, it may be placed on the meta-level as it provides underlying support for both of them.

5 Conclusion

This paper presented an inceptive formal theory for dialogue management that was able to support the discussion on how the chatbot issues can be approached both when examining mechanical dialogues and exploring design decisions; leading to useful insights. This suggests that the theory succeeded at the proof of concept level. In further works, this proposal requires relevant extensions to properly handle a fully developed chatbot.

References

1. Bertalanffy, L. *General system theory: foundations, development, applications.* G. Braziller, 1968.
2. Colby, K., Watt, J., and Gilbert, J. A computer method of psychotherapy: preliminary communication. *The Journal of Nervous and Mental Disease 142*, 2 (1966), 148–152.
3. da Costa, N. *Introdução aos Fundamentos da Matemática.* HUCITEC, 1992.
4. Dewdney, A. K. Computer recreations. *Scientific American 260*, 6 (1989), 122–125.
5. Dijkstra, E. *On the Role of Scientific Thought.* Springer, New York, NY, 1982, pp. 60–66.
6. Ferrara, E., Varol, O., Davis, C., et al. The rise of social bots. *CACM 59*, 7 (2016), 96–104.
7. Fourault, S. The ultimate guide to designing a chatbot tech stack., 2017.
8. Jurafsky, D., Martin, J., and P. Norvig, S. R. *Speech and Language Processing.* Pearson, 2014.
9. Kleene, S. *Introduction to Metamathematics.* Bibliotheca Mathematica. Wolters-Noordhoff, 1952.
10. Levy, H. Gartner predicts a virtual world of exponential change, 2016. viewed: Jun, 2018.
11. Mohammad, S., and Turney, P. Emotions evoked by common words and phrases: Using mech. turk to create an emotion lexicon. In *Proceedings of the CAAGET '10* (2010), Assoc. for Comp. Linguistics.
12. Newell, A. The knowledge level. *Artif. Intell. 18*, 1 (Jan. 1982), 87–127.
13. Parnas, D. The real risks of artificial intelligence. *Commun. ACM 60*, 10 (Sept. 2017), 27–31.
14. Rumbaugh, J., Jacobson, I., and Booch, G. *Unified Modeling Language Ref. Man.* Pearson, 2004.
15. Sedgewick, R., and Wayne, K. *Algorithms.* Pearson Education, 2014.
16. Sørensen, M., and Urzyczyn, P. *Lectures on the Curry-Howard Isomorphism.* Elsevier, 2006.
17. Turing, A. Computing machinery and intelligence. *Mind 59* (1950), 433–460.
18. Weizenbaum, J. Eliza — a computer program for the study (...). *CACM 9*, 1 (Jan. 1966), 36–45.