# A Chatbot to Support Basic Students Questions

Rafael Santana[1], Saulo Ferreira[1], Vitor Rolim[3], Péricles Miranda[1], André Nascimento[1] and Rafael Ferreira Mello[1,2]

[1]Departamento de Computação, Universidade Federal Rural de Pernambuco, Brazil
[2]Cesar School, Brazil
[3]Centro de Informática, Universidade Federal de Pernambuco, Brazil

### Abstract

Chatbots are tools that use artificial intelligence to simulate a human conversation. They can be used for different applications such as providing customer service within an e-commerce or answering FAQs (Frequently Asked Questions). This work proposes the development of a chatbot to help students from a Brazilian public university in the search for information related to the university's administrative processes and general questions about its course. The developed system is able to deliver a high accuracy in the classification of the question's intention and have user answers in a wide range of different topics.

### Keywords

Chatbot, Learning Analytics, Natural Language Processing

## 1. Introduction

Entering higher education causes a series of changes in a student's life. Because the university environment is an unprecedented context in their life, the student can feel lost in the midst of so many stress factors. This situation can become more serious with the lack of guidance about bureaucracy and processes that often narrows down to the department level, and which is usually one of the biggest obstacles in a young person's adaptation to university [1, 2]. Educational institutions can use Information and Communication Technologies (ICT) to reduce the impact of this problem on the student's academic life.

On the other hand, chatbots have been evolving and conquering a space that has long been outdated: customer service. chatbot applications have been disseminated in different areas of knowledge, whether for service in online stores [3], medical assistants [4], assistive technology tool [5] or educational assistants [6]. A chatbot can recognize a certain range of sentences present in its database and formulate instant responses for a large number of users. This type of application enables the modernization of the information supply process, providing agility in resolving doubts and possible problems.

*IV LATIN AMERICAN CONFERENCE ON LEARNING ANALYTICS - 2021. Klinge Villalba-Condori Jorge Maldonado-Mahauad.*

jrafaeldesantana@gmail.com (R. Santana); saulolucas@gmail.com (S. Ferreira); vbr@cin.ufpe.br (V. Rolim); pericles.miranda@ufrpe.br (P. Miranda); andre.camara@ufrpe.br (A. Nascimento); rafael.mello@ufrpe.br (R. F. Mello)
0000-0002-5668-6975 (R. Santana); 0000-0000-0000-0000 (S. Ferreira); 0000-0002-5023-2972 (V. Rolim); 0000-0002-5767-7544 (P. Miranda); 0000-0002-9333-3212 (A. Nascimento); 0000-0003-3548-9670 (R. F. Mello)

Dealing with the current state-of-the-art, how can we add more to the experience of an automatic response system dealing especially with the academic sphere? Relying on this type of questioning prevents us from following the wide availability of chatbots for customer service in marketplaces and bringing benefits related to our context [7, 8].

Given this context, this work proposes to develop a chatbot application, called S.A.N.D.R.A., to help students of a Brazilian public university to have access to administrative information simply and directly. This application must cover information about the university's bureaucratic processes and be able to answer the most common questions that a freshman student may have in their first semesters of the course. Knowing the breadth of topics that the system must cover, how to deal with the amount of different information that can decrease the system's performance? Among these questions, the most recurrent are those related to the time and place of the discipline, internship processes, complementary activities, and general questions such as access to university portals, such as the academic system and the Virtual Learning Environment. The results indicate that S.A.N.D.R.A. achieves an accuracy of approximately 80% in the quality of the response sent.

## 2. Related Work

This section aims to list chatbot applications in the educational context. Several chatbots have already been developed for the education area, among them, we can highlight the CVChatbot [9]. The authors identified a deficiency in the communication between teachers and students using the Virtual Learning Environment (VLE). This problem is usually caused by the students' lack of habit in checking their institutional email inbox or the lack of verification of notifications from virtual rooms on the platform itself. Through integration with Moodle, CVChatbot can improve the communication process between students and teachers, sending messages to students through Facebook Messenger whenever there is a new update in the virtual rooms in which he is enrolled. Through an evaluation questionnaire carried out by the authors, it was found that approximately 72% of the interviewed students consider the use of CVChatbot useful in the communication process between teachers and students.

Chatbots can also be used to facilitate the learning process. In [10] a chatbot project is presented that helps students in the Computer Architecture discipline, answering the most frequent questions asked by students. The bot provides an interface that allows teachers to register, modify, and delete questions and answers from the question database, making it easier to maintain bot knowledge. This chatbot takes away from the teacher the tiring task of always answering the same questions from the students and allows the student to have autonomy during their learning process by not having to wait for a response from the teacher when there is any doubt. The authors did not carry out a detailed assessment of chatbot effectiveness or adherence.

Another example of an educational chatbot is the UNIBOT [11]. UNIBOT can answer several questions related to students and the functioning of the educational institution. UNIBOT uses expressions in SQL queries to find an appropriate answer. However, this technique only provides relevant answers to the user if the user's question and the question registered in the database contain the same words, not recognizing synonymous words. This condition makes this chatbot

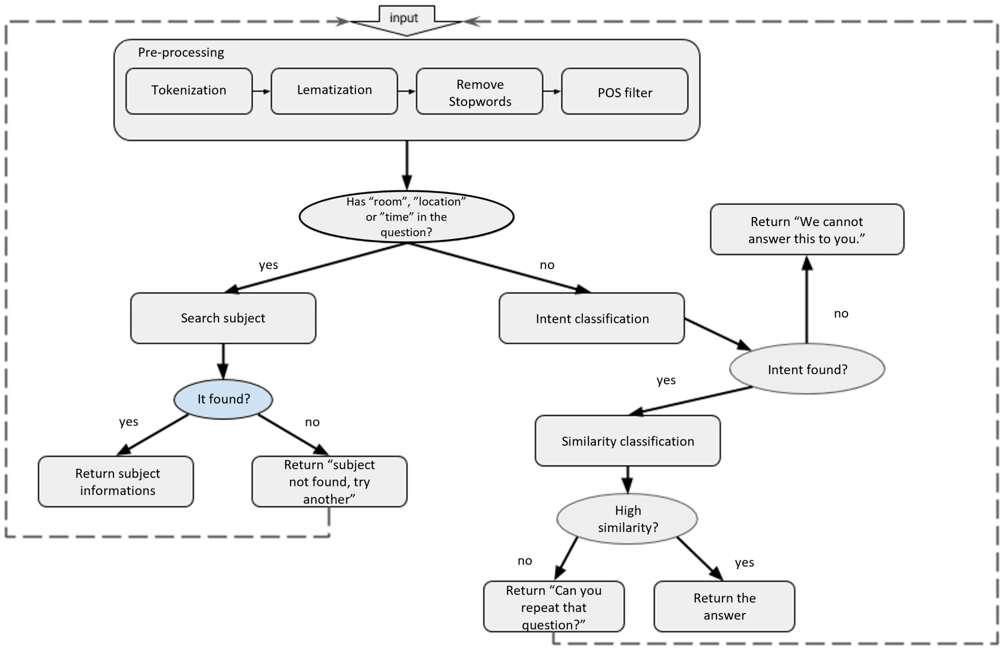


**Figure 1:** Chatbot processing flowchart.

not get a satisfactory result. This chatbot was developed to allow, with few changes, any institution can implement the chatbot on their institutional website. The project is web-based and offers a simple graphical interface similar to a messaging application.

This work proposes the creation of S.A.N.D.R.A., a chatbot that uses Natural Language Processing (NLP) techniques to help resolve possible doubts that a student may have related to university bureaucratic processes, general doubts about the course, and class schedules. The use of NLP allows the chatbot to understand the natural language and extract meaning from the questions, allowing for greater assertiveness in the answers provided.

## 3. Chatbot development process

S.A.N.D.R.A. was developed using some sequential steps necessary to associate a user question with an equivalent stored in the database. This process involves steps such as pre-processing the text, classifying the intention of the user's question, and only then using the database to obtain the necessary information. Classifying the question's topic in advance reduces the margin of error when searching for an answer, as the system will only treat the data that is within your topic.

The flowchart shown in Figure 1 describes the system's internal procedure so that, after the user asks a question, the most appropriate answer is checked. The processing done in each of these steps is intended to allow the user to ask the same question in different ways and the system understands and forwards to the same answer. The benefit of this is to avoid situations where the database has the answer to a question, but the system does not understand the question because it was asked with a variation in writing or sentence structure. The steps

**Table 1**
The number of questions available in each class.

| Class | Initial number of questions | Final number of questions |
|---|---|---|
| Additional activities | 16 | 48 |
| Subjects | 21 | 63 |
| Enrollment | 16 | 48 |
| Internship | 8 | 24 |
| General | 31 | 93 |
| Total | 92 | 276 |

of the flowchart will be detailed below.

### 3.1. Questions dataset

For the training of the chatbot proposed in this work, about 92 questions were extracted from the "Frequently Asked Questions" areas of the websites of the computing courses of a Brazilian public university. These questions were manually divided into topics by related subjects and, for each question, two additional questions were created, to enrich the training database and assess the similarity of the answers, as shown in Table 1.

For the treatment of time/place of subjects, a subject database provided by the university was used, which contains the main information of all 1.453 subjects offered in the semester, including the class, timetables, whether it is mandatory or optional, the department and the building where it will be taught.

### 3.2. Pre-processing

An important step before calculating similarity or classifying text is to pre-process the received data to reduce noise and select only the most relevant. The pre-processing flow consists of tokenization, lemmatization, POS (Part-of-Speech) filtering, and stopwords removal, in that order. The use of these techniques is common in text classification works, such as [12, 13, 14], which analyze the importance of text pre-processing before applying classification algorithms.

Tokenization at the beginning of pre-processing is performed so that we can treat the received data at the word level, not more than a sentence. The data is divided into words called *tokens* and through it, we can apply the other pre-processing steps. Like lemmatization, a process that transforms inflected words into their root form, it is applied so that the word is recognized as one, regardless of its variation in gender, number, degree, or verb conjugation.

The POS filtering is used to select the only relevant part of speech for classification. Conjunctions, articles, and pronouns are classes common to all question sentences and do not help to identify a specific question or question class. Like stopwords, neutral words that are irrelevant because they are used frequently in most texts but do not contribute to the recognition of the context of a sentence or text.

### 3.3. Subject search

The S.A.N.D.R.A. was developed to serve undergraduate courses at a public university that uses a database with a list of subjects and their respective information, such as time and place of classes. Subject names were also pre-processed to facilitate matching with user-provided information. By identifying keywords ("time", "room" or "location"), the system processes the subsequent words, or *tokens*, to find some combination between these and the disciplines available in the database.

To avoid that some errors in the writing of the subject are an obstacle for the name of the subject received to coincide with any subject in the database, a syntactic margin was established using the distance of Levenshtein [15], developed for comparison of *strings* is widely used for DNA sequence analysis [16] and optimized grammar correction [17]. The subject that is closest to the one sent by the user is selected, that is if it obeys a minimum similarity degree to be chosen, which we define as 5.

### 3.4. Intent classification

An intent is a related subject topic, where various question types can be mapped, and grouping them into these topics is a way of narrowing down the questions that will be selected for calculating similarity to the question received by the chatbot. In the database used, the questions were separated into some topics that were frequently repeated so that each question received could be classified among the different intentions.

Questions about enrollment, discipline, complementary activities, and internships are common in academic environments. These topics were transformed into our intentions, that is, the question, before being assimilated to an answer, will be paired with a set of questions associated with your topic. Also added the "general" topic, for questions that don't fit these previous ones but are equally recurrent. Different methods of intention classification were used, in a classical way, and using state-of-the-art concepts, these classifiers are described in the next subsections.

#### 3.4.1. Classic classifiers

Using classic artificial intelligence classifiers implemented by *scikit-learn* [18] with its default parameters, the first step is to transform the sentence into a list of scalars. For this, it is common to use the TF-IDF (Term Frequency-Inverse Document Frequency) [19] method, which calculates the weight of a word by its recurrence in documents, to estimate its relevance for classifying a sentence as belonging to a particular topic. This method is widely used because it not only considers the frequency of a term in a specific document but its frequency in a set of documents. This starts from the idea that a very frequent term in a document may be important, but a very frequent term in all documents may just be a common term.

Obtaining the scalar vectors of the sentence received as input, the system uses Support Vector Machines (SVM) [20], Decision Tree [21] and Random Forest [22] to classify the sentence into one of the defined topics, each topic being a list of pre-processed questions and transformed into scalar vectors.

To classify our sentence, SVM uses *kernels* functions (in our case, RBF) to map the possibly non-linearly separable data into a linearly separable dataset and find a line capable of dividing the groups between different classes, in our case, intentions.

In Decision Tree, the algorithm seeks to create a tree based on rules, performing increasingly homogeneous divisions in the data, where the number of classes is increasingly smaller. This process is done to classify the data based on the purest subgroup obtained, that is, a new object following a certain set of rules is associated with a purer group of objects where most of the classes present should be its final classification.

In addition, Random forest uses combined Decision Tree models for better results. In this process, the algorithm randomly obtains subsets of the object's feature vector and selects the best features from these groups. This procedure usually obtains better results due to the diversity obtained in the extraction of features.

#### 3.4.2. Deep learning

Another way to classify intent using state-of-the-art methods is using the DIET (Dual Intent Entity Transformer) [23], a multi-tasking architecture used for both intent classification and entity recognition.

This architecture uses pre-trained embedding models integrated with BERT [24], GloVe [25] and ConveRT [26] to get better ranking results.

This representation by embeddings is often used because it tries to map the semantic value of the phrase into multidimensional scalar vectors. These vectors seek to describe the meaning of words, making it possible to extract feelings and calculate semantic similarity with other words. To generate them, co-occurrence matrices and probabilistic methods are used along with neural networks. The purpose is that words with similar meanings have close vector representations.

### 3.5. Similarity analysis

After classifying the intention, the number of questions to be compared is restricted to those pertaining to the given topic. These questions are transformed into scalar vectors using the TF-IDF, as in the intention classification, as well as the input provided by the user. With the input sentence and the topic questions in scalar vector format, the cosine distance was used to associate the user's question with the corresponding question in the database. Obeying a certain minimum similarity, initially defined as $0.7$, the answer to the most similar question is selected to be returned to the user.

To validate this method and compare the TF-IDF approach with another approach using the Euclidean distance between word embeddings, we performed the tests with our question database. For each question in the database, two similar ones were created to be classified within the group of questions for their topic. Thus, there are 184 questions to be tested against the initial 92.

## 4. Results

During the development of the S.A.N.D.R.A., several approaches were tested to measure the accuracies achieved using our database of frequently asked questions. To assess the generaliz-

**Table 2**
Results of validating algorithms for class intent extraction.

| | Additional activ. | Subjects | Internship | General | Enrollment | Mean |
|---|---|---|---|---|---|---|
| Random Forest | 0.5000 | 0.4107 | 0.3818 | 0.5636 | 0.4727 | 0.4657 |
| SVM | 0.7142 | 0.8392 | 0.8181 | 0.7454 | 0.5272 | 0.7288 |
| Decision Tree | 0.7500 | 0.8035 | 0.8363 | 0.7090 | 0.5636 | 0.7325 |
| DIET | 0.9642 | 0.9268 | 0.8333 | 0.9047 | 0.9310 | 0.9162 |

**Table 3**
Result of validation of similarity calculation approaches.

| | Hits | Errors | Ratio |
|---|---|---|---|
| TF-IDF + cosine distance | 146 | 38 | 0.7934 |
| Embeddings + euclidean distance | 123 | 61 | 0.6684 |

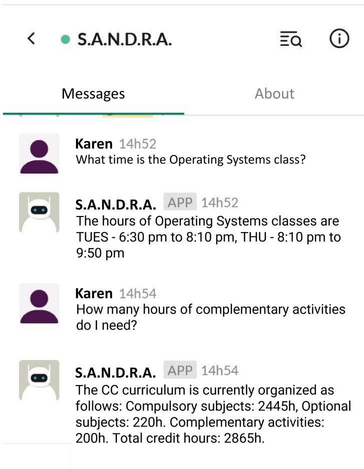


**Figure 2:** Question answer.

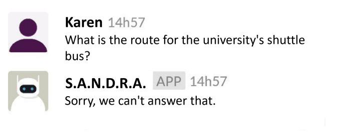


**Figure 3:** Error message.

ability of our models, the cross-validation technique K-Fold [27] was used, where 5 *folders* were used.

The Table 2 shows that DIET obtained the best result in predicting intention in almost all categories, losing only in the category “Internship” for the decision tree algorithm. This result is more evident when we compare the final mean of the accuracy of all approaches. The DIET obtained an accuracy approximately 25% greater than the decision tree, in second place.

The validation of similarity analysis methods using the extra questions elaborated and the initial database was one of the experiments and obtained the results presented in Table 3. The validation shows that the approach that uses TF-IDF together with the cosine distance achieved a superior result than the approach using embeddings and euclidean distance. Finally, Figures 2 and 3 present some examples of student interactions with the S.A.N.D.R.A.

## 5. Discussion

Using a deep learning model as part of the ranking increased hit rates even with a limited database. The pre-trained embedding models added information to the FAQ database questions. The results obtained by the DIET model greatly increase the accuracy of this process compared to classical machine learning algorithms that use only our database data.

Intent classification restricts the number of data to be analyzed and compared, solving the initial problem of the breadth of topics that the chatbot must address. By ranking the question on a subject in common with other questions in the database, our similarity ranking model only handles related question data.

The applied pre-processing was successful in extracting the content of the questions, contributing to the high precision in the comparison of sentences. You can analyze this point with the high hit rate in our similarity calculation. Because of this, the system can recognize different ways of asking the same question and map these into an equivalent answer, covering more questions that can be asked.

In addition to questions about university information, our system also handles questions related to subjects, using word processing methods to return to the user the discipline information present in our database even if the user does not enter their name rigidly equal. This makes it possible for the user to more easily obtain the time and place of the subject they want, a point that the related works did not address.

## 6. Conclusions and future work

With the advancement of technology and research on artificial intelligence, especially in the area of Natural Language Processing, it became possible to create conversation robots that are closer to human language. The chatbots provide a high level of service efficiency, support if there is a high demand for services and a decrease in operating costs.

This work presented the S.A.N.D.R.A., a chatbot to help new students with the administrative processes of the university. The chatbot can answer the most frequently asked questions a student may have in their first few months at university. In addition to the benefit to the student, the course coordinators are also benefited by avoiding the repetitive task of always answering the same questions at the beginning of the semester.

A limitation for the S.A.N.D.R.A. it's the lack of ability to store the context of the conversation, treating each user's message as a single message, and ignoring the history of previous messages. This issue is noticeable when a user asks a question based on some previous message. When trying to reply to this new message, the chatbot does not take the previous message into account.

To evaluate the effectiveness of the chatbot, a response evaluation system can be implemented in which the user can assign a score to each response sent by the robot, where later the chatbot administrator can improve those that are poorly evaluated. An administrative panel can also be developed to facilitate the addition, editing, and deletion of questions present in the chatbot database, avoiding a long manual process whenever a question is added or modified.

## References


[1] B. C. P. de Brito, R. F. L. de Mello, G. Alves, Identificação de atributos relevantes na evasão no ensino superior público brasileiro, in: Anais do XXXI Simpósio Brasileiro de Informática na Educação, SBC, 2020, pp. 1032–1041.

[2] E. Freitas, T. P. Falcão, R. F. Mello, Desmistificando a adoção de learning analytics: um guia conciso sobre ferramentas e instrumentos, Sociedade Brasileira de Computação (2020).

[3] A. Bhawiyuga, M. A. Fauzi, E. S. Pramukantoro, W. Yahya, Design of e-commerce chat robot for automatically answering customer question, in: 2017 International Conference on Sustainable Information Engineering and Technology (SIET), 2017, pp. 159–162.

[4] P. Srivastava, N. Singh, Automatized medical chatbot (medibot), in: 2020 International Conference on Power Electronics IoT Applications in Renewable Energy and its Control (PARC), 2020, pp. 351–354.

[5] C. da Silveira, A. R. da Silva, F. Herpich, L. M. R. Tarouco, Uso de agente conversacional como recurso de aprendizagem sócio-educacional, RENOTE-Revista Novas Tecnologias na Educação 17 (2019).

[6] A. J. Sinclair, R. Ferreira, D. Gašević, C. G. Lucas, A. Lopez, I wanna talk like you: Speaker adaptation to dialogue style in l2 practice conversation, in: International Conference on Artificial Intelligence in Education, Springer, 2019, pp. 257–262.

[7] R. Ferreira-Mello, M. André, A. Pinheiro, E. Costa, C. Romero, Text mining in education, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 9 (2019) e1332.

[8] A. P. Cavalcanti, A. Diego, R. Carvalho, F. Freitas, Y.-S. Tsai, D. Gašević, R. F. Mello, Automatic feedback in online learning environments: A systematic literature review, Computers and Education: Artificial Intelligence (2021) 100027.

[9] P. Dehon, A. Silva, A. Inocêncio, C. Castro, H. Costa, P. Júnior, Cvchatbot: Um chatbot para o aplicativo facebook messenger integrado ao ava moodle, Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE) 29 (2018) 1623. URL: https://www.br-ie.org/pub/index.php/sbie/article/view/8123. doi:10.5753/cbie.sbie.2018.1623.

[10] F. A. Mikic-Fonte, M. Llamas-Nistal, M. Caeiro-Rodríguez, Using a chatterbot as a faq assistant in a course about computers architecture, in: 2018 IEEE Frontiers in Education Conference (FIE), 2018, pp. 1–4.

[11] N. P. Patel, D. R. Parikh, D. A. Patel, R. R. Patel, Ai and web-based human-like interactive university chatbot (unibot), in: 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), 2019, pp. 148–150.

[12] V. Srividhya, R. Anitha, Evaluating preprocessing techniques in text categorization, International journal of computer science and application 47 (2010) 49–51.

[13] A. Krouska, C. Troussas, M. Virvou, The effect of preprocessing techniques on twitter sentiment analysis, in: 2016 7th International Conference on Information, Intelligence, Systems & Applications (IISA), IEEE, 2016, pp. 1–5.

[14] M. Toman, R. Tesar, K. Jezek, Influence of word normalization on text classification, Proceedings of InSciT 4 (2006) 354–358.

[15] V. Levenshtein, Leveinshtein distance, 1965.

[16] L. P. Dinu, A. Sgarro, A low-complexity distance for dna strings, Fundamenta Informaticae

73 (2006) 361–372.
[17] K. U. Schulz, S. Mihov, Fast string correction with levenshtein automata, International Journal on Document Analysis and Recognition 5 (2002) 67–85.
[18] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.
[19] K. S. Jones, A statistical interpretation of term specificity and its application in retrieval, Journal of documentation (2004).
[20] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, B. Scholkopf, Support vector machines, IEEE Intelligent Systems and their Applications 13 (1998) 18–28.
[21] L. Breiman, J. Friedman, C. J. Stone, R. A. Olshen, Classification and regression trees, CRC press, 1984.
[22] T. K. Ho, Random decision forests, in: Proceedings of 3rd international conference on document analysis and recognition, volume 1, IEEE, 1995, pp. 278–282.
[23] T. Bunk, D. Varshneya, V. Vlasov, A. Nichol, Diet: Lightweight language understanding for dialogue systems, 2020. arXiv:2004.09936.
[24] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).
[25] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1532–1543.
[26] M. Henderson, I. Casanueva, N. Mrkšić, P.-H. Su, T.-H. Wen, I. Vulić, Convert: Efficient and accurate conversational representations from transformers, 2019. arXiv:1911.03688.
[27] B. Efron, Estimating the error rate of a prediction rule: Improvement on cross-validation, Journal of the American Statistical Association 78 (1983) 316–331. URL: http://www.jstor.org/stable/2288636.