Chatbot as support to decision-making in the context of natural resource management*

Bruno C. Alves[2], **Larissa A. de Freitas**[2] **and Marilton S. de Aguiar**[1,2]

[1]Graduate Program in Computer Science
[2]Technological Development Center
Federal University of Pelotas – Pelotas – RS – Brazil

{bcalves,larissa,marilton}@inf.ufpel.edu.br

Abstract. *The management of natural resources is becoming increasingly relevant due to its direct implication in society's life. Thus, individuals must make decisions based on environmental and social aspects. This work uses a chatbot to support users' decisions through an RPG scenario based on the participatory management of resources in the Lagoa Mirim Watershed and Canal São Gonçalo Basin. In this context, in addition to the chatbot, this study presents a pollution predictor to support decision-making, with a determination coefficient of 0.99, constructed using random forest. Also, we present five Word Embeddings models to expand the natural language understanding, based on a corpus of about 700 thousand sentences, capable of identifying relations between words.*

1. Introduction

Natural resource management is an area that seeks better ways to manage land, water, plants, and animals, based on the quality of life in society. This area has gained visibility for governments due to sustainable development, which is a principle of how they see and understand the world. Natural resource management has specific objectives the scientific study of resources and how these resources can support life [Holzman 2009]. Water is one of the most important natural resources, as it is essential for social and economic activities [Ponte et al. 2016]. The management of water resources involves different groups and organizations, which need to analyze better ways of distributing and using water.

Considering that this resource is shared and limited, decision-making is a relevant aspect for this management because it is possible to obtain more appropriate solutions through the interaction between individuals [Adamatti 2007]. In this context, Machine Learning (ML) represents systems from computational tools to support risk prediction. Considering the growing presence of chatbots in everyday life [Raj 2019], this type of communication, based on natural language, presents as an alternative for the information propagation that helps in the implementation of actions. Thus, extracting meaning from messages sent to the chatbot can be applied to Natural Language Processing (NLP) and constructed vector representation models with Word Embedding (WE).

The development of this work is in the context of an in-progress research project. In this project, a computational game based on Multiagent Systems (MAS)

*This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES/Brasil) and Agência Nacional de Águas (ANA/Brasil) – Edital 16/2017.

and Role-Playing Game (RPG) for the natural resource management is being implemented, more specifically for the participatory management of water resources in the Lagoa Mirim Watershed and Canal São Gonçalo, located in the south of Brazil. In the RPG [Leitzke et al. 2019], called GORIM, players interpret characters within a story constructed through rules, modeled after interactions with the region's hydrographic basin committee, where players make decisions and communicate with other characters (agents) searching for their individual/collective goals.

Regarding this context, this work presents the development of a chatbot capable of assisting different RPG roles in decision-making. For example, considering environmental information and interactions between characters in a watershed scenario, game agents can consult trends using statistics and make predictions about pollution levels based on a model constructed with ML, applied to the data collected in the RPG pilot sessions. Also, we developed five WEs models to expand the chatbot understanding through the vector representation of words since the resources available in the literature for application in NLP tasks in the Portuguese language are limited.

We organized this article as follows. In Section 2, we present the theoretical background for this work; in Section 3, we describe the technical/methodological decisions that guided the development of this study; in Section 4, we discuss the results obtained; and finally, in the Section 5, we present the conclusions of this work.

2. Theoretical Background

This Section will present concepts about Chatbots, ML, NLP, and WE areas in the context of this study. Besides, this Section will discuss the main related works.

A chatbot (also referred to as a conversational agent) is an automated program that seeks to answer questions based on the simulation of human behavior [Raj 2019]. Researchers developed chatbots of various technologies for different purposes in areas such as commerce, school, and health from this event. Currently, there are specific platforms for structuring conversational agents including Watson Assistant[1], Wit.ai[2], and Dialogflow[3]. All of these applications use NLP, so it becomes possible to implement and integrate chatbots. According to the complexity of the algorithms used in their construction, we can classify conversational agents based on rules or self-learning. In the rules-based method, the chatbot seeks to answer questions asked according to a set of simple specifications. In contrast, in the self-learning strategy, machine learning techniques are used during conversational agent training [Hussain et al. 2019].

ML can be applied for automatic data analysis to obtain helpful knowledge that assists in resource management and decision-making. In particular, predictive models, based on previous experiences (supervised learning), can be constructed from regression models, which seek to extract patterns from the data and thus predict continuous values [Alpaydin 2014]. Therefore, to understand this work, four regression algorithms will be presented: i) linear regression, this algorithm is an equation that describes the relationship between a dependent variable and a set of attributes; ii) support vector regression

[1] https://www.ibm.com/cloud/watson-assistant

[2] https://wit.ai/

[3] https://dialogflow.cloud.google.com/

(SVR), we use this algorithm for seeking a maximum margin that separates the hyperplane to gather the most significant number of data in a narrow area; iii) regression tree, this algorithm is a set of rules based on predictive attributes; and, iv) random forest regression, this algorithm calculates the average of the predictions of a group of regression trees.

NLP is an area composed of techniques that seek to extract meaning from the human's natural language, such as English and Portuguese. Usually, developers apply NLP techniques in chatbots proposed to solve tasks involving the self-learning method, to simplify and standardize the raw text [Eisenstein 2019]. Thus, the main NLP techniques involved are: i) normalization – this technique adequacy the text in terms of spelling, removal of accents, and removal of special characters; ii) tokenization – this technique separates the text in individual terms called tokens; iii) removal of stopwords – this technique removes the words with little relevance; and, iv) lemmatization – this technique transforms the verbs to the infinitive and adjectives/nouns to the masculine singular.

WEs are the texts converted into a numerical representation turning it possible to map the words of a group of texts (corpus) into real low-dimension vectors, making it possible to capture semantic aspects of the terms [Lane et al. 2019]. For this work, we use the following approaches of representation at the word level: i) Word2Vec, this algorithm represents words based on the training of neural networks, being able to perform the analysis considering context words (CBOW) or just a word (Skip-gram) [Mikolov et al. 2013]; ii) GloVe, this algorithm extracts the meaning of the terms from the proportions of the probabilities of co-occurrences of tokens and global characteristics of the corpus [Pennington et al. 2014]; and, iii) FastText, based on Word2Vec, represents words through the sum of the learning obtained by n-gram character sets [Bojanowski et al. 2017].

In [Sawant et al. 2019], the authors proposed a random forest classifier for predicting the best harvest season, associated with a chatbot implemented in Dialogflow. In [Nallappan 2018], a system was created for cost prediction with the use of the Statistical Model ARIMA. Finally, in the work [Kannagi et al. 2018] a tool for predicting yields in harvests was described using algorithms such as linear regression and SVR. The methods for understanding the chatbots of the last two works consist of classic NLP methods of high dimension and cannot handle semantic tasks. Thus, this work differs from the other ones by treating the resource management applied to an RPG, presenting a predictor model of pollution for the environment of a watershed. In addition, for a better understanding of the chatbot, five WE models were created.

3. Proposed Approach

This Section will present the proposed system, specifying the pollution predictor model and the natural language understanding model. As shown in Figure 1, users can send audio and text messages. The server responsible for the system receives the messages and sends them to the Dialogflow API, which processes the information and searches for an appropriate response format. If there is no correspondence, the server will request the natural language understanding API, composed of NLP and WE techniques, to search for a similar question in a repository. Also, to complement the answer, external searches are performed in a database or the pollution predictor API, generated from regression algorithms. Finally, the user receives the complete response through text and graphics.

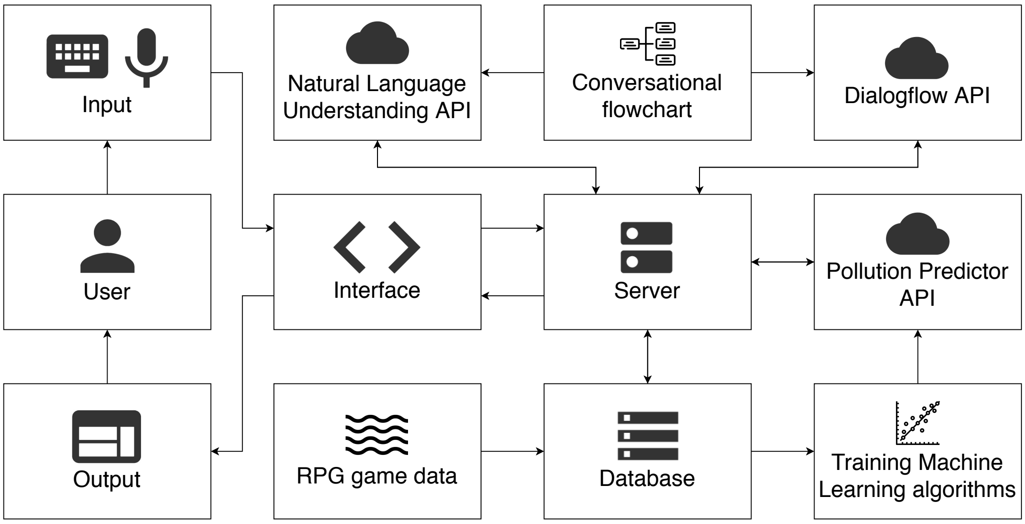
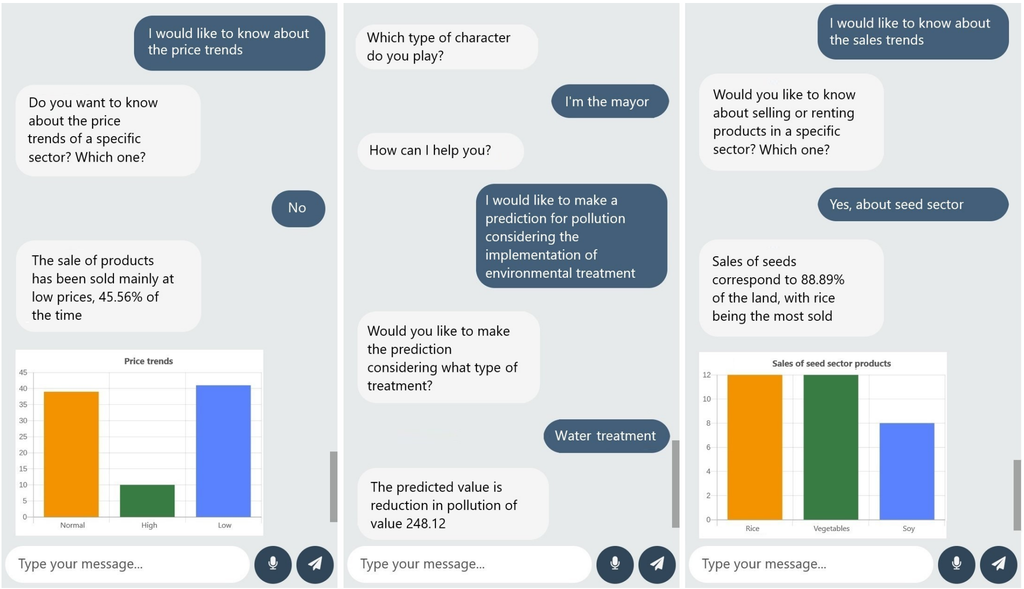


Figure 1. Architecture proposed for the chatbot.

Figure 2 illustrates the system interface, with examples of dialogues related to price trends, pollution prediction for the mayor using the model generated with ML algorithm, and information about seed sales.


I would like to know about the price trends

Do you want to know about the price trends of a specific sector? Which one?

No

The sale of products has been sold mainly at low prices, 45.56% of the time

Which type of character do you play?

I'm the mayor

How can I help you?

I would like to make a prediction for pollution considering the implementation of environmental treatment

Would you like to make the prediction considering what type of treatment?

Water treatment

The predicted value is reduction in pollution of value 248.12

I would like to know about the sales trends

Would you like to know about selling or renting products in a specific sector? Which one?

Yes, about seed sector

Sales of seeds correspond to 88.89% of the land, with rice being the most sold

Figure 2. The system interface presenting examples produced by the chatbot.

3.1. Pollution Predictor Model

As presented in Section 1, the RPG studied in this work has the natural resource management as scenario. This scenario allows interactions between agents where the characters must follow the rules according to their roles in the game environment.

Regulators are agents who act as public people, managing the financial resources obtained through the application of taxes paid by society. With these resources, the mayor and the alderman can discuss and implement pollution control policies. *Supervisors* are agents who inspect/report irregularities related to the environment exploration. The NGO (Non-Governmental Organization) is responsible for reporting environmental conditions to regulators. The inspector is responsible for inspecting the producers and penalizing them if they violate the regulators' rules. *Producers* are agents who explore the environment to obtain financial resources, including the farmer and the businessman. The interaction between these agents occurs through the purchase/rent and sale of equipment and supplies. Therefore, the businessman agent makes products available for the production of the farmer agent.

The actions of each agent provide the data for implementing predictive models. These records were collected directly from the game engine during eight-game simulations, totaling 34 rounds between 2019 and 2020. Unfortunately, the original dataset was not in a suitable format for use. It was necessary to restructure the information in a new dataset, according to the following steps: importing logs, creating columns, and calculating costs, balances, productivity, and pollution per action. Finally, to store this data was created an SQL database.

In the RPG, each player can perform only actions compatible with their role, and each act can affect the ecosystem. It is possible to measure this through pollution, which reflects the impact of actions on RPG. Thus all agents can impact the environment. However, it is possible to achieve a balance by implementing environmental treatment, tax adjustments, and conscious actions by producers/supervisors. Regarding this context, a pollution predictor model was constructed with records stored in the database, totaling 3763 lines representing an action involving one or two agents. In addition to the target attribute, 11 predictive attributes were considered, related to the type of action executed, two possible types of agents involved in the transaction and their respective balances, products sold/rented and their respective price, environmental treatment, green seal, and values of fines and taxes.

Lately, we pre-process the dataset to adjust missing values with zeros and convert categorical data into numeric ones. It was considered the StratifiedKFold cross-validation method for the separation of data between training and testing. Based on this method, implemented in the Scikit-Learn[4] library, the algorithm carried out ten iterations with the data divided into ten groups, so each of these iterations refers to the set with test data and the rest to the training data. Considering the constructed dataset, we trained four regression algorithms using Python language and Scikit-Learn library. Linear regression, SVR, regression tree, and random forest regression are all in the context of supervised learning, as presented in Section 2. Therefore, the parameterization of the linear regression, SVR (regularization and epsilon parameters corresponding to 1 and 0.1, respectively), and regression tree followed the pattern proposed by the library. However, for the random forest regression, 50 trees were defined because there is no significant improvement with values greater than this. After training the algorithms, we obtained the results presented in Section 4. Thus, the predictor regression model with the best performance was implemented in an API, with a Flask[5] framework, considering the data used in this work.

[4] https://pypi.org/project/scikit-learn/

[5] https://pypi.org/project/Flask/

3.2. Natural Language Understanding Model

Considering the game modeling proposal, we elaborate conversational flows about relevant questions for the agents involved in the RPG environment. Thus, individual flowcharts were developed on the Dialogflow platform, comprising conversations related to the prediction on pollution levels, in addition to dialogues based on statistics, such as prices and sales, according to data from the game engine. For this work, we use Dialogflow to structure the base dialogs because, in addition to having similar aspects to other platforms in the area, it has a free use license.

To expand the understanding of the chatbot, we constructed a system based on NLP and WE to select an adequate response to users. For this purpose, it was necessary to use a set of texts that contain aspects of natural resources in the Portuguese language. Thus, we use the corpus collected by [Drury et al. 2017] during the experiments. This corpus contains about 97 thousand news about the agricultural area from 1997 to 2016. Furthermore, the author provides a WE model for the Word2Vec algorithm through document-level training. Therefore, for this study, we constructed all WEs models through analysis at the sentence level, in addition to NLP techniques.

We converted the annotated raw texts from the corpus into about 700 thousand sentences. Following that, we process the sentences with the support of NLP techniques, included in the discussion of Section 2: normalization, tokenization, removal of stopwords, and lemmatization. Thus, after implementing these techniques, the sentences are written in lowercase letters, organized as a set of relevant words in their lemma format. For this task, we use the spaCy[6] and NLTK[7] NLP libraries.

Subsequently, we map the words into numeric vectors by representing five models of WEs: Word2vec (CBOW and Skip-gram), GloVe, and FastText (CBOW and Skip-gram). We trained all these models with the lemmas, 50 dimensions, and 50 epochs/iterations. Besides, we use 10 for the context window because, according to [Miñarro-Giménez et al. 2015], this can have a loss of performance when using a greater number. We used Gensim[8] for training the Word2Vec and the FastText models, and Glove_Python[9] for training the GloVe model.Finally, we created a Flask API to search answers equivalent to the questions sent by users. In this API, the message sent by an agent is processed using the NLP techniques mentioned above and converted into numerical weights using the WEs models. Thus, the system compares the distance between user sentence vectors and repository sentence vectors for each of the models through cosine similarity. Based on this measure, it is possible to determine the similarity between sentences according to the vectors' orientation's proximity. Therefore, the similarity between the vectors is high when they are close. We use a voting system between the results obtained by WEs models to return the output sentence, corresponding to the input sentence, with the highest number of votes to the user.

[6] https://pypi.org/project/spacy/

[7] https://pypi.org/project/nltk/

[8] https://pypi.org/project/gensim/

[9] https://pypi.org/project/glove_python/

4. Results and Discussions

This Section will present the results of this study. Thus, we will discuss the model developed for pollution prediction through regression and the model constructed for expanding the understanding of chatbot using WEs. Four regression models for pollution were generated, according to the development presented in Section 3.1. For this study, we considered three metrics for evaluating regression algorithms: mean absolute error (MAE), mean squared error (MSE), and coefficient of determination (R^2 Score). According to applied metrics, the lower the MAE and MSE, the better the regression is represented. In contrast, R^2 Score returns a maximum value equal to one (best case), based on the MSE value and the variance.

Table 1. Results of metrics applied to regression models.

Algorithm	MAE	MSE	R^2 Score
Linear Regression	0.7044	6.7336	0.9902
Support Vector Regression (SVR)	0.4576	8.0922	0.9883
Decision Tree	0.4027	9.8278	0.9858
Random Forest Regression	**0.3948**	**6.5115**	**0.9907**

Table 1 summarizes the results, considering these metrics in a scenario of 30 experiments. According to this table, the best results refer to the random forest regression and linear regression algorithms. Also, we observed that the models generated from the regression tree and SVR obtained minor successes in predictive tasks, especially when considering the MSE. Pondering the metrics, we determine that the random forest model was the algorithm with the best performance. Thus, it is consistent that the random forest has generated the best model when using a set of regression trees. Regarding the WEs models[10], we created five types of representations to apply in NLP tasks in Portuguese. Furthermore, as presented in Section 3.2, we proposed a proportional voting system among the five models to search for the most similar sentence in a repository. Thus, the system compared 200 sentences based on the conversational flows to verify the algorithms' participation during the choice process. In general, an 85% correspondence rate was obtained by the majority through the voting system, disregarding ties.

Table 2. Participation of Word Embeddings models by number of votes.

Model	Two votes	Three votes	Four votes	Five votes	Total
Word2vec (CBOW)	8	25	27	104	164
Word2vec (Skip-gram)	11	24	29	104	168
GloVe	11	19	28	104	162
FastText (CBOW)	8	22	22	104	156
FastText (Skip-gram)	10	25	29	104	168

Table 2 shows the participation of WEs models in the voting of correct sentences. When observing the data in this table, it appears that in 52% (104 sentences), all the WEs models agreed about the proper determination of the most similar sentence, generating a

[10] https://github.com/brunocascaes/WordEmbedding

total of five votes. Regarding the sentences in which the algorithms had more difficulty in agreeing (in the elections won by two votes), we observed that the Word2Vec (Skip-gram) and Glove models presented the best involvement in 11 sentences. In general, Skip-gram models were present during the most successful choices in 168 sentences. However, we use all five models during the choosing because there is a wide variety of words. We use the t-SNE[11] dimensionality reduction algorithm to generate illustrations for the WEs models since the implemented models have 50 dimensions. Therefore, t-SNE reduced the dimensions of vectors by two-dimensional points represented by the axes "x" and "y". In this way, similar terms are modeled by close points, preserving the relations between words. Considering that there are thousands of words, it is impossible to view all names and relations in just one graphic. Thus, we chose five terms related to the study area: "agricultural", "environmental", "plantation", "pollution", and "reservoir".

Figure 3 shows five clusters, based on each of the five words mentioned above, for the Word2Vec, FastText, and GloVe models. Through these figures, it is possible to observe the relationships between the groups composed of 30 terms, in addition to the two most related words found by the models for each term. When analyzing these representations, we observed that the models based on FastText have, in general, their clusters better divided because they analyze words using n-grams of characters. An important point to note is that, for this reason, these models may be most susceptible to capture some noise in the corpus, as is the case of "agricultural" visualized in the graphic of the FastText (CBOW). However, given that users make types, noise capture by FastText models is considered relevant. In contrast, models like Word2Vec and GloVe present a wide diversity of words in common, such as the related correspondence between "agricultural" and "fertilizer" visualized in the graphic of the Word2Vec (Skip-gram). Also, we observed that all five models constructed of WE could be used for the proposed task. Thus, it appears that the models based on FastText capture most aspects related to the structure of words. While the models of Word2Vec and GloVe find different terms with close meanings, however not necessarily similar in writing. Therefore, the results of the models proved to be adequate. In particular, when we use the models in a group, they can handle various linguistic tasks, increasing the ability to understand the relationships between words.

5. Conclusion

In this work, we proposed a chatbot to support decision-making in the natural resource management of an RPG game based on the Lagoa Mirim Watershed and Canal São Gonçalo Basin environment. Thus, we generated a conversational agent to assist users through information and a pollution predictor model. For this model, we trained four ML algorithms using data from the game engine. When analyzing the results, the model that obtained the best performance was the random forest regression with an R^2 Score of 0.9907. Also, we proposed a natural language understanding model that searches for the most similar sentence in a repository. For this, we used NLP techniques and WEs models combined with cosine similarity. In this scenario, based on a corpus of about 700 thousand sentences, five models were trained by WE: Word2vec (CBOW and Skip-gram), GloVe, and FastText (CBOW and Skip-gram). With these models, it became possible to identify the relations between words, including aspects of the word substructures, less-used terms,

[11] https://pypi.org/project/tsne/

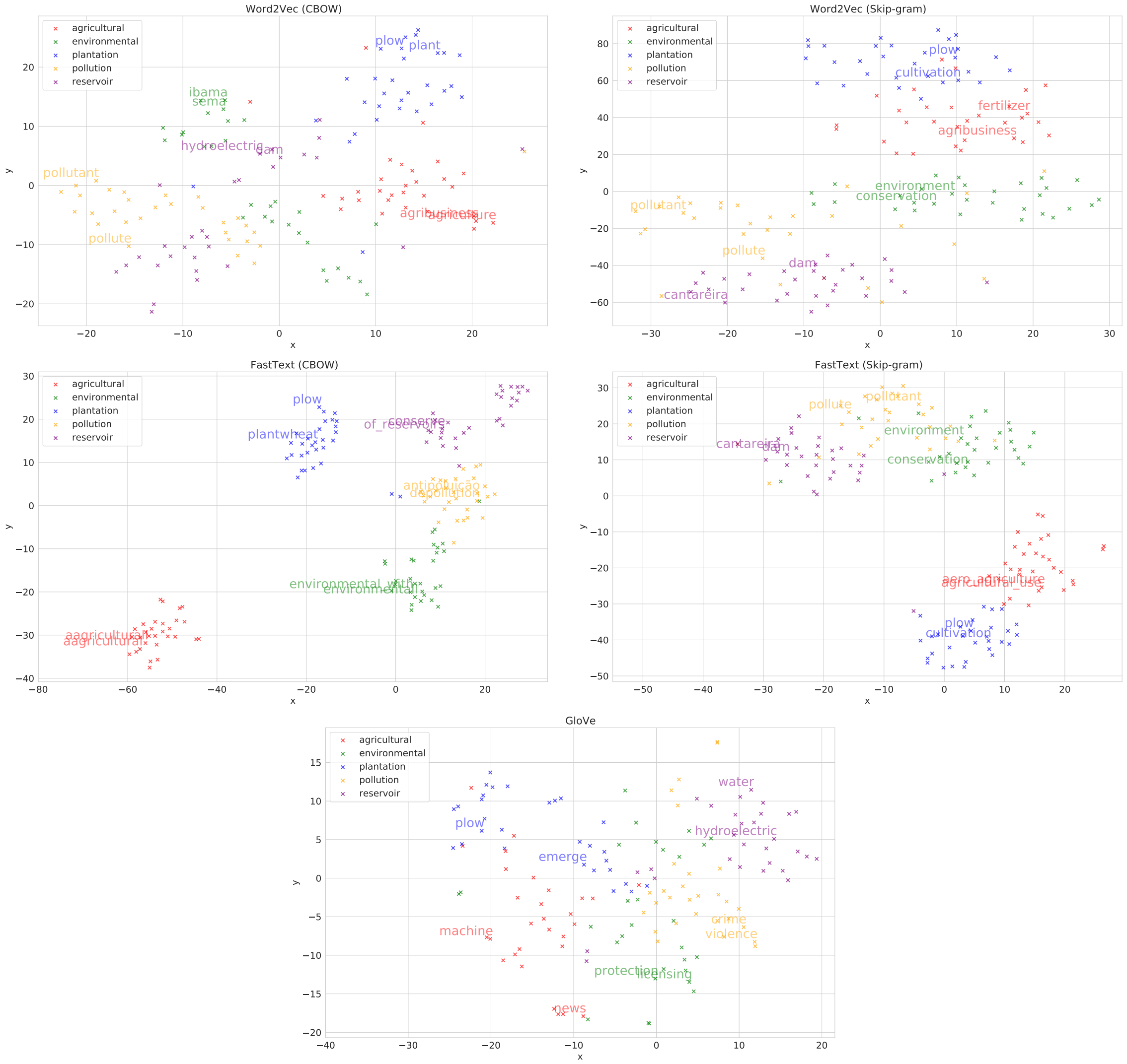


Figure 3. Representation of Word Embeddings models for five clusters of words.

and context words. Considering the wide variety of words, we defined a voting system between the WEs models when choosing the sentence with the highest meaning level. In the context of water resources management, this research remains a relevant topic of study because it is possible to analyze human behavior and support their decisions based on a conversational agent through the interaction between RPG players in the management simulation of natural resources.

References

Adamatti, D. F. (2007). *Inserção de jogadores virtuais em jogos de papéis para uso em sistemas de apoio à decisão em grupo: um experimento no domínio da gestão de recursos naturais*. PhD thesis, Escola Politécnica, Universidade de São Paulo, SP.

Alpaydin, E. (2014). *Introduction to Machine Learning*. MIT Press, Cambridge, MA, USA, 3 edition.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Lin-*

guistics, 5:135–146.

Drury, B., Fernandes, R., and Lopes, A. (2017). Bragrinews: Um corpus temporal-causal (português-brasileiro) para a agricultura. *Linguamática*, 9.

Eisenstein, J. (2019). *Introduction to Natural Language Processing*. Adaptive Computation and Machine Learning series. MIT Press, Cambridge, MA, USA.

Holzman, B. (2009). Natural resource management. [Online; accessed 18 fev. 2021] `http://online.sfsu.edu/bholzman/courses/GEOG%20657/`.

Hussain, S., Sianaki, O., and Ababneh, N. (2019). *A Survey on Conversational Agents/Chatbots Classification and Design Techniques*, pages 946–956. Springer International Publishing, Cham, DE.

Kannagi, L., Ramya, C., Shreya, R., and Sowmiya, R. (2018). Virtual conversational assistant:'the farmbot'. *International Journal of Engineering Technology Science and Research*, 5(3):520–527.

Lane, H., Howard, C., and Hapke, H. (2019). *Natural Language Processing in Action*. Manning Publications, New York, NY, USA.

Leitzke, B., Farias, G., Melo, M., Gonçalves, M., Born, M., Rodrigues, P., Martins, V., Barbosa, R., Aguiar, M., and Adamatti, D. (2019). Sistema multiagente para gestão de recursos hídricos: Modelagem da bacia do são gonçalo e da lagoa mirim. In *Workshop de Computação Aplicada a Gestão do Meio Ambiente e Recursos Naturais*, pages 87–96, Porto Alegre, RS, Brasil. SBC.

Mikolov, T., Chen, K., Corrado, G. S., and Dean, J. (2013). Efficient estimation of word representations in vector space.

Miñarro-Giménez, J. A., Marín-Alonso, O., and Samwald, M. (2015). Applying deep learning techniques on medical corpora from the world wide web: a prototypical system and evaluation.

Nallappan, M. (2018). A prediction system for farmers to enhance the agriculture yield using cognitive data science. *International Journal of Advanced Research in Computer Science*, 9:780–784.

Pennington, J., Socher, R., and Manning, C. (2014). GloVe: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Ponte, B., de la Fuente, D., ParreÑo, J., and Pino, R. (2016). Intelligent decision support system for real-time water demand management. *International Journal of Computational Intelligence Systems*, 9(1):168–183.

Raj, S. (2019). *Building Chatbots with Python: Using Natural Language Processing and Machine Learning*. Apress, New York, NY, USA.

Sawant, D., Jaiswal, A., Singh, J., and Shah, P. (2019). Agribot - an intelligent interactive interface to assist farmers in agricultural activities. In *Proceedings of the IEEE Bombay Section Signature Conference (IBSSC)*, pages 1–6, Mumbai, India. IEEE.