Signal2Vec: Time Series Embedding Representation

Christoforos Nalmpantis(✉) and Dimitris Vrakas

School of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece
{christofn,dvrakas}@csd.auth.gr
https://www.csd.auth.gr/en/

Abstract. The rise of Internet-of-Things (IoT) and the exponential increase of devices using sensors, has lead to an increasing interest in data mining of time series. In this context, several representation methods have been proposed. Signal2vec is a novel framework, which can represent any time-series in a vector space. It is unsupervised, computationally efficient, scalable and generic. The framework is evaluated via a theoretical analysis and real world applications, with a focus on energy data. The experimental results are compared against a baseline using raw data and two other popular representations, SAX and PAA. Signal2vec is superior not only in terms of performance, but also in efficiency, due to dimensionality reduction.

Keywords: Time series · Data mining · Representations · Time series classification · Energy embeddings · Non intrusive load monitoring

1 Introduction

Time series is a sequence of data in time order, with values in continuous space. The order can be irrelevant to time, but it is still important. This type of data has always attracted the interest of scientists in a vast range of areas such as speech recognition, finance, physics, biology etc. Some common tasks involving time series are: motif discovery, forecasting, source separation, subsequence matching, anomaly detection and segmentation.

In time series problems, regardless the approach, the performance of the solution is heavily affected by the representation of the data. The categories of representations can be classified into data adaptive, non-data adaptive, model-based

This work has been funded by the ΕΣΠΑ (2014–2020) Erevno-Dimiourgo-Kainotomo 2018/EPAnEK Program 'Energy Controlling Voice Enabled Intelligent Smart Home Ecosystem', General Secretariat for Research and Technology, Ministry of Education, Research and Religious Affairs.
We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

J. Macintyre et al. (Eds.): EANN 2019, CCIS 1000, pp. 80–90, 2019.
https://doi.org/10.1007/978-3-030-20257-6_7

and data dictated. The first one includes techniques such as Adaptive Piecewise Constant Approximation [13], Singular Value Decomposition [15], Symbolic Natural Language [23], Symbolic Aggregate ApproXimation [16]. Approaches, which belong to the second representation, are: Discrete Wavelet Transform [5], spectral DFT [9], Piecewise Aggregate Approximation [14] and Indexable Piecewise Linear Approximation [6]. Model based representations are based on statistics such as Markov Models and Hidden Markov Model [19] and Auto-Regressive Moving Average [7]. Finally, the most popular data dictated approach is Clipped [24].

In this paper a novel framework, named Signal2vec, is introduced. A similar approach has been proposed by Nalmpantis et al. [20], where a model called Energy2vec is used to create a hyperspace of energy embeddings. Energy2vec is binded to the energy domain, is supervised and its applicability is limited. On the other hand, Signal2vec is a general, unsupervised model and is applicable in any time series. It is inspired by Word2vec [17] which builds a vector space, maintaining semantic and syntactic relations of the original words. Word2vec has been applied on numerous textual or discrete sequences such as recommendation systems [2,22], ranking of sets of entities [4,10], biology [1] and others [26]. Signal2vec is the first attempt, that extends Word2vec applicability on any sequential data in continuous space.

The benefits and the drawbacks of the framework are discussed extensively through a theoretical analysis. The framework is validated with experiments in two different tasks: classification and single source separation. Both, the analysis and the experiments are based on energy data.

2 Signal2Vec

Signal2vec consists of two main steps: tokenization and skip-gram model. The former one is a discretization process, transforming a continuous time series into tokens. The latter one transforms the sequence of tokens into embeddings. Figure 1 illustrates the steps of the framework.

2.1 Tokenization

Unsupervised tokenization is an abstract and scalable approach in order to discretize a time series. It can be applied on different domains and it can fit different variations of data in the same domain. It is completed in two main steps: token extraction and token assignment. The first step can be achieved by a clustering algorithm. The second one uses classification, in order to transform a continuous time series to a sequence of tokens.

At the current implementation, k-means is used to define the tokens, the number of which is not known upfront. The best number of tokens, which is also the number of clusters, is found by using silhouette score, within a desirable range of values. Silhouette score shows which objects lie well within their cluster [25] and the desirable range of values can be defined empirically. In energy domain this range can be estimated by calculating the number of possible energy states

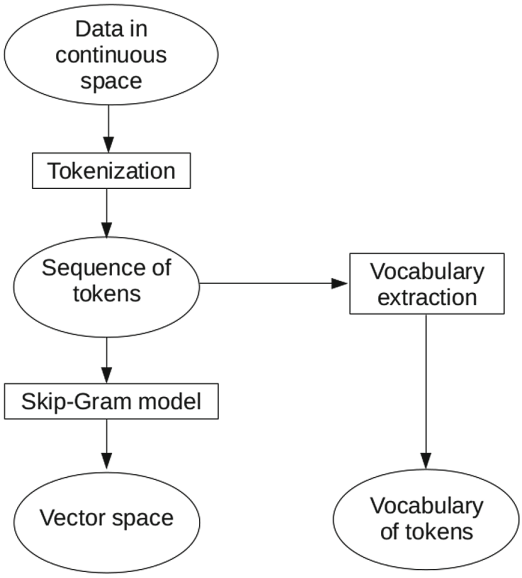

Fig. 1. Signal2Vec framework.

using complexity of power draws [8]. Next, the classifier k-nearest neighbors is trained to map values of the signal to tokens. The classifier can be used to tokenize time series from the same domain.

The algorithm is evaluated using household energy data. Tokens represent the energy states of each appliance, because token extraction is applied on the submetered data. Then, sequences of tokens are created for each appliance. The final sequence is a concatenation of the appliance specific sequences and corresponds to the aggregated signal. In problems like power disaggregation, both token extraction and token assignment would be applied directly on the aggregated signal, because the submetered data are supposed to be unknown. The analysis that follows is based on submetered data, in order to present meaningful tokens, that correspond to appliances states.

2.2 Skip-Gram

Signal2vec is based on word2vec, which uses either skip-gram model or continuous bag-of-words (CBOW). Skip-gram predicts the words around the target word and CBOW predicts the target word given its neighbours. Both methods can be applied having minor differences on the results. For consistency, skip-gram is selected for all the experiments.

Following tokenization, a time series is mapped to a sequence of tokens. This sequence is now called a corpus. If the tokens are not abstract states and reflect real-world conditions of the time series, then the corpus is a human description of the total signal. The collection of the tokens consists a vocabulary. In order

to apply the skip-gram model, a context is also defined as the window to the left and to the right of the target token. The objective of the algorithm is to predict the context given a specific token. The architecture is a shallow neural network with one hidden layer and is trained with pairs of tokens. One token is the target and the other one belongs to the context. The network trains the weights of the hidden layer from the frequencies each pairing shows up. A more formal definition of the objective function is defined next.

Let tkn_1, tkn_2,...tkn_T be a sequence of T training tokens. Then the objective function tries to maximize the average log probability according to the formula:

$$1/T \sum_{t=1}^{T} \sum_{-c \leq_i \leq_c, i \neq_0} \log p(tkn_{t+i}|tkn_t), \tag{1}$$

where c is the training context.

The order of the tokens in a context doesn't affect the result of the algorithm, which depends mainly on the frequency of the tokens. In word2vec model this is mentioned by Mikolov et al. [18] as a limitation of the model, because it cannot capture rules that dictate the order of words in a sentence. In time series usually there aren't any syntactic rules and there is no difference if a token is before or after its neighbors.

The network is trained computing Noise Contrastive Estimation (NCE) [11] loss function. The optimizer is the Adagrad with learning rate 0.001. The size of each embedding is 300 and the window is 6 tokens. The data come from House 1 of UK-DALE dataset [12] during the year 2014.

3 Evaluation

3.1 Data and Tools

The source of the data is the UK-DALE dataset, which includes both the aggregated and the individual power consumption of the appliances in a house. House 1 is selected, because it has the most devices of all the houses. The preferred programming language is Python. The tool named NILMTK [3] is used for accessing the database and preprocessing the energy data. In order to distribute computation to many CPU cores and a GPU, the skip-gram model is developed in Tensorflow. Tensorflow comes along with a suite of visualization tools, called Tensorboard. It is used to plot diagrams of the model, visualize the embeddings and evaluate the model. Tensorboard's visualization tool uses PCA and TSNE in order to plot the embedding space in three or two dimensions. The evaluation of the embedding space is mainly done with tensorboard's similarity tool, which supports both cosine and euclidean distance.

3.2 Analysis of the Learned Representations

Signal2vec is a framework which transforms a time series to a continuous vector space. In order to understand the intuition of a signal's geometrical representation, an evaluation process is presented, focusing on household energy data.

In unsupervised tokenization the tokens are defined in an abstract mathematical way. Tokenizing the aggregated energy signal is helpful solving real world problems, without any insight about the nature of the tokens. In this evaluation the method is applied on the individual signals, extracting multiple states per appliance. No window is used and tokens are mapped in high resolution, close to the sampling rate. In order to get a physical image of the energy behavior, diagrams depicting the frequency of each energy state are generated.

The name of the states is labeled by the name of each appliance, followed by a number. Thus, the name of an energy state doesn't directly reveal the real world functionality, but only which appliance it belongs to. Also, regarding the zero state of an appliance, it is no more distinguishable from the other states, as it is just another extra label with arbitrary number. A diagram of an appliance which is ON continuously would be the same with a diagram of an appliance being continuously OFF. To avoid any ambiguity, energy plots from raw data are used, as well.

Tensorboard is used to visualize and explore the embedding space. Dimensionality reduction is achieved by means of PCA. The geometry of the space has two distinct groups of points. One group represents high frequency states and the other one low frequency states. The same separation based on frequency is seen in word embeddings, where words follow Zipf's law.

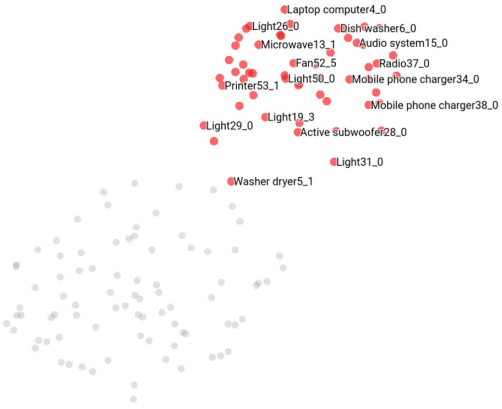

Fig. 2. TSNE: The embeddings are grouped in two clusters.

Another useful tool, which comes with Tensorboard, is a clustering analysis using TSNE. The majority of the combinations of the values of perplexity and learning rate give clear separation of the two groups that have been identified with PCA. Figure 2 shows an example of the TSNE algorithm with perplexity 5 and learning rate 10. The two clusters are consistent even when trying different settings of the algorithm. Only the shape is changing when the perplexity number is much bigger. For example, perplexity 40 gives a circle, when projecting the

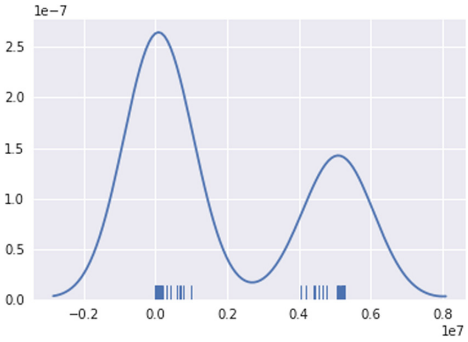

Fig. 3. Frequency distribution of tokens.

embeddings in 2D and the two groups are gathered in two different semicircles. In 3D the clusters are more clear because the two semicircles are separated as two distinct shapes.

The similarity tool is used to find vectors that are close to each other. Both Euclidean and cosine distance metrics are used, although there isn't any significant difference in the results. For each similarity search, the respective plots of the input vector and its five closest ones are compared. The results show that embeddings with small distance in the geometrical space, have similar frequency diagrams. The results are very robust in terms of distinguishing high and low frequency energy states. Almost all the cases of similarity searches give neighbor vectors, which correspond to the same category of frequency. On the other hand, vectors belonging to the same cluster cannot be distinguished and no characteristics are found to justify the results of a similarity search.

Figure 3 depicts the frequency distribution of tokens, derived from unsupervised tokenization. There are two central points, forming two normal distributions. Comparing the frequency distributions and the geometric properties of the embedding spaces, there is a connection between the tokens and the embeddings. Skip-gram, transfered the properties of the sequence of tokens to a multidimensional space. Assuming that sequences of tokens can be translated to frequency of appliance usage, which in turn implies human behaviour and habits, it can be concluded that the constructed vector space encapsulates the energy profile of the house.

4 Real World Applications

In this proposal two real world experiments are presented, classification and energy disaggregation. The first one examines the capabilities of the proposed framework to classify signals from different appliances efficiently. The second one is a simple approach on how a single source separation problem, such as energy disaggregation, can be solved using Signal2vec. The data are from UK-DALE house 1 during the year 2014. The difference is that now the vectors are

produced from the aggregated energy signal and they are used to transform any other energy time series.

4.1 Multiclass Classification of Appliances' Energy Consumption

The first experiment is a multiclass classification problem, identifying 12 different appliances: oven, microwave, dish washer, fridge freezer, kettle, washer dryer, toaster, boiler, television, hair dryer, vacuum cleaner and light. The classifier is a random forest with 200 estimators. The dataset of different labeled signals is created as follows. Firstly, the submetered data of 12 appliances are converted to sequences of 300 dimensional vectors, using Signal2vec. Next, the data are broken into smaller pieces with fixed length, corresponding to a specific time period. For example during 9 months with data sampled every 6s, a time period of 1 day and 12 appliances would give approximately 4188 labeled sequences of vectors. Then the average vector of each 1 day length time series is calculated. The average vector is the input to the classifier and the label is one of the 12 appliances. The metric that is used is mainly macro f1-score. The robustness of the results was validated using a k-fold cross-validation, after the initial data had been randomly shuffled. The parameter k is chosen as $k = 3$, because for larger values the test data sample was not statistically representative of the broader dataset.

The results vary depending on the size of each time series. The smaller the time period is, the worse the results are. This can be explained because in periods smaller than a day some devices are not used at all. Indeed, for devices that are used daily, such as fridge freezer, the classifier could recognize the appliance successfully even with a time period of 4 h with individual f1-score 0.95. The respective macro f1-score for the 12 appliances was 0.39. Consequently, the experiments for 12 appliances are meaningful for time period greater than a day. Another approach would be to have an extra label for time periods during which no device is on, known as zero state, but this is left for future work.

The framework is robust when the appliances are increased. For example using 7 day length time series for the cases of 6, 12 and 17 appliances the macro f1-score is 0.97 (+/− 0.019), 0.94 (+/− 0.015) and 0.79 (+/− 0.015) respectively. Figure 4 shows a confusion matrix summarizing the classification results of 12 appliances. It is worth mentioning that most of the misclassifications concern the oven. This is not a surprise because an oven's energy consumption heavily depends on the way it is used.

Table 1 compares Signal2vec against raw data and two other representations, SAX [16] and PAA [14]. The results involve time series with sizes 1, 5 and 7 days, using 3 cross validation. Other classifiers have also been tested. Regardless the representation, random forest showed the best results. SAX and PAA have been tuned to get the best possible f scores. Overall Signal2vec is superior, not only giving the best results, but also because the dimension of the final representative vector is independent of the length of the time series. Signal2vec is surpassed by PAA, only for short time series such as 1 day size. A possible explanation is that temporal patterns are more important during short periods, because a device is

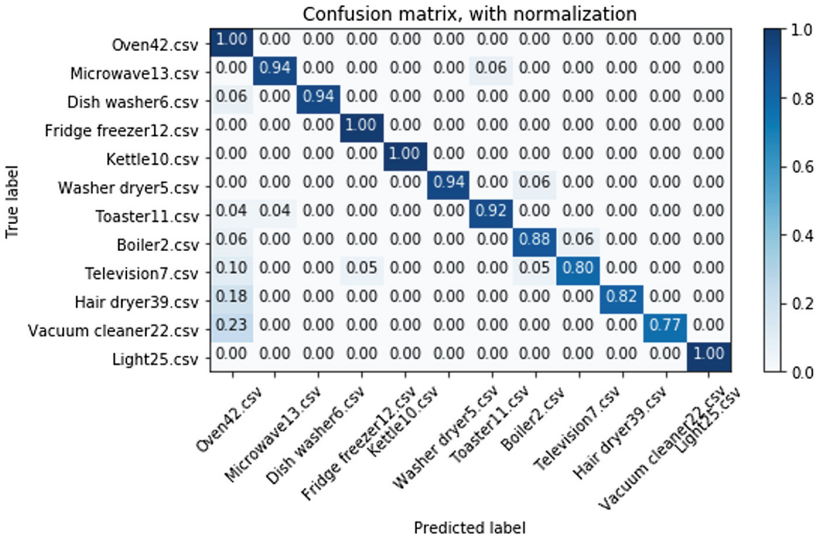

Fig. 4. Confusion matrix of 12 classes using Signal2vec.

used rarely. When calculating the average vector any temporal patterns are lost, whereas in PAA they are maintained. For future experiments more sophisticated methods can be evaluated e.g. weighted average vector. Finally, it is notable that for larger time series the results for raw data are getting worse. This is explained by the dimensionality explosion, which signal2vec faces by compressing all the information into a single vector of constant dimensions.

Table 1. Macro f1 score for 12 appliances.

TS size	Signal2vec	Raw	SAX	PAA
1 day	0.730	0.730	0.682	**0.814**
3 days	**0.878**	0.732	0.748	0.869
5 days	**0.930**	0.681	0.767	0.897
7 days	**0.942**	0.693	0.766	0.920

4.2 Energy Disaggregation

Many machine learning approaches, including deep learning, have been proposed for the problem of energy disaggregation [21]. The majority of them use one model for each appliance, because they underperform when trying to disaggregate many devices. The current experiment tackles energy disaggregation as a multilabel classification problem. The input data come from the main aggregated energy signal, following the same procedure as in the first experiment. The whole

time series during 2014 is segmented into fixed windows. Signal2vec is applied to each window and finally the average vectors are calculated. Each representative vector is the input to a multilayer percepton classifier with one hidden layer and 100 neurons. The fixed windows that have been tested correspond to 4, 8, 12 and 24 h. The labels are the appliances that were ON at least one time during the fixed time period. The results are very encouraging, especially when comparing the performance of the same model identifying 6 and 12 appliances. Table 2 presents the results in details. Additional experiments need to be implemented for comparison with other models.

Table 2. Macro f1 score of energy disaggregation.

TS size	6 Appliances	12 Appliances
4 h	0.534 (±0.029)	0.512 (± 0.008)
8 h	0.654 (± 0.005)	0.599 (± 0.037)
12 h	0.722 (± 0.015)	0.678 (± 0.043)
24 h	0.868 (± 0.032)	0.757 (± 0.045)

5 Conclusion

Signal2vec is a computationally efficient model, which transforms the data of a time series into a vector space. The trained embeddings are easily reusable and maintain some of the properties of the original data. It is unsupervised, requires no domain knowledge, is scalable and applicable in different areas e.g. speech recognition, finance, health, IoT.

Specifically, in energy domain, it is the first time the energy profile of a building is mapped to a multidimensional space. Once the embeddings are trained, they can be reused either by incorporating them in a neural network or by other models as features. Two different classification problems have been showcased, with application in real world problems. The first one classifies different categories of signals, coming from different sources. The second one, energy disaggregation, is a single source separation problem. Both experiments showed very promising results, reducing a time series of thousands values to a 300 dimensional vector.

Further research is suggested to be conducted to improve Signal2vec, show experimental results in other real world problems and compare against state-of-the-art methods.

References

1. Asgari, E., Mofrad, M.R.: Continuous distributed representation of biological sequences for deep proteomics and genomics. PloS one **10**(11), e0141287 (2015)
2. Barkan, O., Koenigstein, N.: Item2Vec: neural item embedding for collaborative filtering. In: 2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP), pp. 1–6. IEEE (2016)
3. Batra, N., et al.: NILMTK: an open source toolkit for non-intrusive load monitoring. In: Proceedings of the 5th International Conference on Future Energy Systems, pp. 265–276. ACM (2014)
4. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Advances in Neural Information Processing Systems, pp. 2787–2795 (2013)
5. Chan, K.P., Fu, A.W.C.: Efficient time series matching by wavelets. In: Proceedings of the 15th International Conference on Data Engineering, 1999, pp. 126–133. IEEE (1999)
6. Chen, Q., Chen, L., Lian, X., Liu, Y., Yu, J.X.: Indexable PLA for efficient similarity search. In: Proceedings of the 33rd International Conference on Very Large Data Bases, pp. 435–446. VLDB Endowment (2007)
7. Corduas, M., Piccolo, D.: Time series clustering and classification by the autoregressive metric. Comput. Stat. Data Anal. **52**(4), 1860–1872 (2008)
8. Egarter, D., Pöchacker, M., Elmenreich, W.: Complexity of power draws for load disaggregation (2015). arXiv preprint arXiv:1501.02954
9. Faloutsos, C., Ranganathan, M., Manolopoulos, Y.: Fast subsequence matching in time-series databases, vol. 23. ACM (1994)
10. Garcia-Duran, A., Bordes, A., Usunier, N.: Composing relationships with translations. Ph.D. thesis, CNRS, Heudiasyc (2015)
11. Gutmann, M.U., Hyvärinen, A.: Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. J. Mach. Learn. Res. **13**(Feb), 307–361 (2012)
12. Kelly, J., Knottenbelt, W.: The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. Sci. Data **2**, 150007 (2015)
13. Keogh, E., Chakrabarti, K., Pazzani, M., Mehrotra, S.: Locally adaptive dimensionality reduction for indexing large time series databases. ACM Sigmod Rec. **30**(2), 151–162 (2001)
14. Keogh, E.J., Pazzani, M.J.: A simple dimensionality reduction technique for fast similarity search in large time series databases. In: Terano, T., Liu, H., Chen, A.L.P. (eds.) PAKDD 2000. LNCS (LNAI), vol. 1805, pp. 122–133. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-45571-X_14
15. Korn, F., Jagadish, H.V., Faloutsos, C.: Efficiently supporting ad hoc queries in large datasets of time sequences. In: ACM Sigmod Record, vol. 26, pp. 289–300. ACM (1997)
16. Lin, J., Keogh, E., Wei, L., Lonardi, S.: Experiencing SAX: a novel symbolic representation of time series. Data Min. Knowl. Disc. **15**(2), 107–144 (2007)
17. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space (2013). CoRR abs/1301.3781. http://arxiv.org/abs/1301.3781
18. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)

19. Minnen, D., Isbell, C.L., Essa, I., Starner, T.: Discovering multivariate motifs using subsequence density estimation and greedy mixture learning. In: Proceedings of the National Conference on Artificial Intelligence, vol. 22, p. 615. AAAI Press; MIT Press, Menlo Park, Cambridge, London (1999, 2007)
20. Nalmpantis, C., Krystalakos, O., Vrakas, D.: Energy profile representation in vector space. In: 10th Hellenic Conference on Artificial Intelligence SETN 2018. ACM (2018)
21. Nalmpantis, C., Vrakas, D.: Machine learning approaches for non-intrusive load monitoring: from qualitative to quantitative comparation. Artif. Intell. Rev. 1–27 (2018)
22. Ozsoy, M.G.: From word embeddings to item recommendation (2016). arXiv preprint arXiv:1601.01356
23. Portet, F., et al.: Automatic generation of textual summaries from neonatal intensive care data. Artif. Intell. **173**(7–8), 789–816 (2009)
24. Ratanamahatana, C., Keogh, E., Bagnall, A.J., Lonardi, S.: A novel bit level time series representation with implication of similarity search and clustering. In: Ho, T.B., Cheung, D., Liu, H. (eds.) PAKDD 2005. LNCS (LNAI), vol. 3518, pp. 771–777. Springer, Heidelberg (2005). https://doi.org/10.1007/11430919_90
25. Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math. **20**, 53–65 (1987)
26. Wu, L., Fisch, A., Chopra, S., Adams, K., Bordes, A., Weston, J.: Starspace: Embed all the things (2017)! arXiv preprint arXiv:1709.03856