How Graph Neural Networks help to analyze the temporal evolution of the French Presidential campaign on Twitter

Keywords: Graph Neural Networks, Dynamic Graph, Online communities evolution, Digital parties

This conference paper aims to highlight the utility of Graph Neural Networks models for computational social sciences. It introduces a new framework using Deep Learning models in order to perform a textual, temporal and network-based study of the structuration of political communities on Twitter.

We focus on the evolution of the French political communities during the 2022 presidential campaign using a Twitter dataset. The scope of this dataset was defined theoretically for the sociological part of this research. In several European countries, traditional political parties are being challenged during presidential campaigns by the emergence of new political communities (i.e. the *Podemos* movement in Spain, the *Five Star Movement* in Italy). Gerbaudo (2018) [1] calls these new political organizations "digital parties" because they have fewer resources than traditional parties and they need to use social media to structure their national and regional networks of supporters. In France, these digital parties are now more important than traditional parties as President Emmanuel Macron created a digital party during the 2017 election, and the most important far left party (*La France Insoumise*) and one of the main far right movements (*Reconquête*) also have a digital party structure. Therefore, this interdisciplinary research evaluates whether social media were used differently during the French 2022 campaign by traditional parties and digital parties (Hypothese 1), especially the rising extreme right-wing parties (Hypothese 2). We therefore use Graph Neural Networks to analyze the uses of Twitter by these different types of political parties and to determine the temporal structuring of these communities during this campaign.

This dataset includes all the tweets that mentioned the names of the main candidates to this election and/or the names of their political parties. It also includes tweets or retweets of the twitter accounts of the candidates as well as a selected group of French journalists covering the elections. In order to analyze the temporal evolution of political communities, these tweets were collected over a period of several months between January 1, 2022 and the second round of the election, in April 2022. This dataset allows us to build a network comprised of 3 531 265 unique users for 82 905 412 interactions (i.e. retweets or mentions).

We first build a dynamic relational network based on the retweets from content related to the election.

Then, in order to scale up our study and overcome the limitations of conventional methods, we learn a representation of this network, i.e. we compute a vector for each user of the network describing him in an embedding space.

To this end, we train a model of Graph Neural Network known as Temporal Graph Network (TGN) [2] to produce a representation that incorporates the structural and the temporal information behind each interaction among users. The neural network is trained by trying to predict future interactions among users based on the embedding it produces.

We also combined the French language model "*Camembert*" [3] and a dimensionality reduction technique to produce an embedding of each tweet and use it as features for the training of our models.

Doing so changes the learned representation and allows us to achieve higher predictive accuracy compared to solely relying on structural and temporal information, thus demonstrating the interest of incorporating textual information.
The learned representation is of interest in several aspects.
It can be used to visualize the communities as users are close in the embedding space if they had a similar online behavior. On top of that, we have at our disposal a time-evolving vector for each user so we can follow the temporal evolution of communities or focus on the trajectories of users of interest inside the embedding space. Another interesting aspect of the representation is that it can be used to perform traditional machine learning on graphs (clustering, label propagation…) with better results. Finally, we can predict the evolution of the network or sub-parts of the networks as the TGN model is trained to perform this task. The first TGN models we trained on a small part of the dataset comprised of 7 543 users and 45124 retweets were able to achieve over 95% accuracy in predicting future retweets (that were not shown to the model during training). The obtained embedding is shown in Figure 5.

Our framework is motivated by experiments we performed on toy data generated using Stochastic Block Models [4]. Following the experiments led by Goyal et al. [5], we implemented a Dynamic version Stochastic Block Model that generates graphs with several communities and then dynamically transfers nodes from one community to another by creating new interactions in a continuous framework. The results of these experiments are shown in Figure 1 to Figure 4.

By training a TGN model on these data, we were able to demonstrate that the model effectively learns the temporal dependencies. Indeed, nodes at the beginning of their transition to a new community are located near the community they will join in the embedding space while still having far less edges with it than with their original community. It also confirms the interest of this method as traditional graph representation algorithms (using spatialization or random walks for instance) fail to discriminate the nodes involved in a community transition from their original community. Ultimately, Graph Neural Networks allows us to have a fine-grained view of the evolution of online political communities and helps to characterize the dynamics at stake in times of elections. This method is particularly useful for analyzing the reconfiguration of political communities during the most crucial phases of presidential elections, for example when supporters of candidates who lost in the first round have to choose the candidate for whom they will vote in the second round.

References

[1] Gerbaudo P., *The digital party: Political organisation and online democracy*, Pluto Press, 2018.

[2] Emanuele Rossi, Ben Chamberlain, Fabrizio Frasca, Davide Eynard, Federico Monti, and Michael Bronstein. Temporal graph networks for deep learning on dynamic graphs, 2020.

[3] Louis Martin, Benjamin Muller, Pedro Javier Ortiz Sua´rez, Yoann Dupont, Laurent Ro- mary,E ricdelaClergerie,Djame´Seddah,andBenoˆıtSagot.CamemBERT:atastyFrench language model.

[4] Stochastic blockmodels: First steps. *Social Networks*, 5(2):109–137, 1983.

[5] Palash Goyal, Sujit Rokka Chhetri, and Arquimedes Canedo. dyngraph2vec: Capturing network dynamics using dynamic graph representation learning. *CoRR*, abs/1809.02657, 2018.

Figure 1 to Figure 4 : Temporal Evolution of the embeddings on Dynamic Stochastic Block Model (DSBM) data obtained with TGN following the nodes changing from community 0 to community 1 (in green). Projection using t-SNE algorithm.

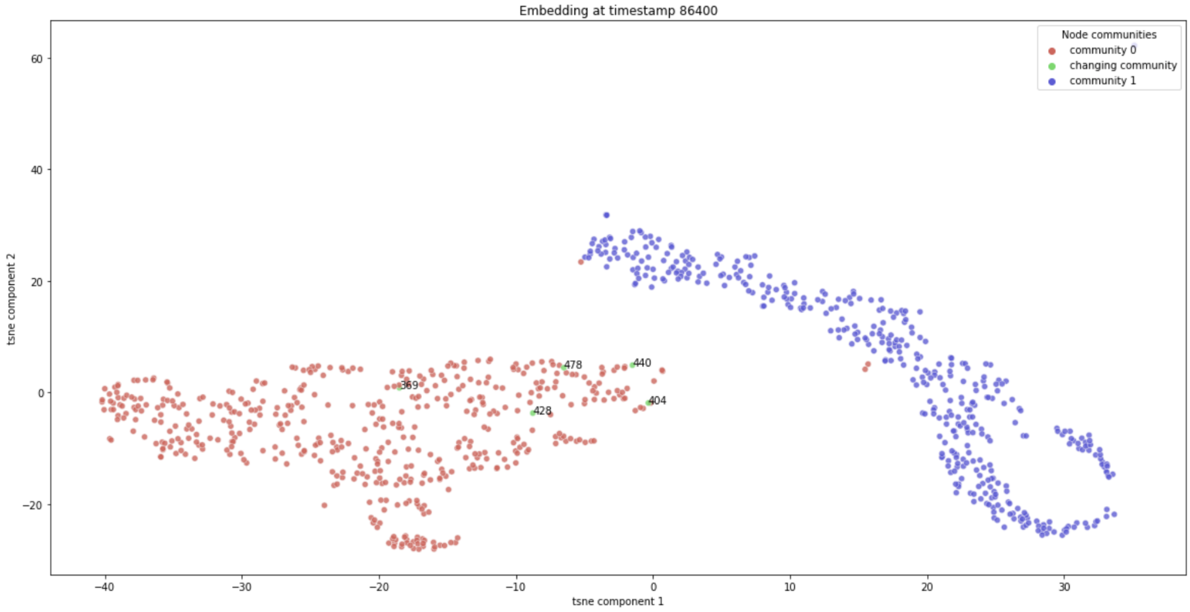

Figure 1: The communities are initialized using DSBM and new interactions are being created between the node changing community and nodes from community 1

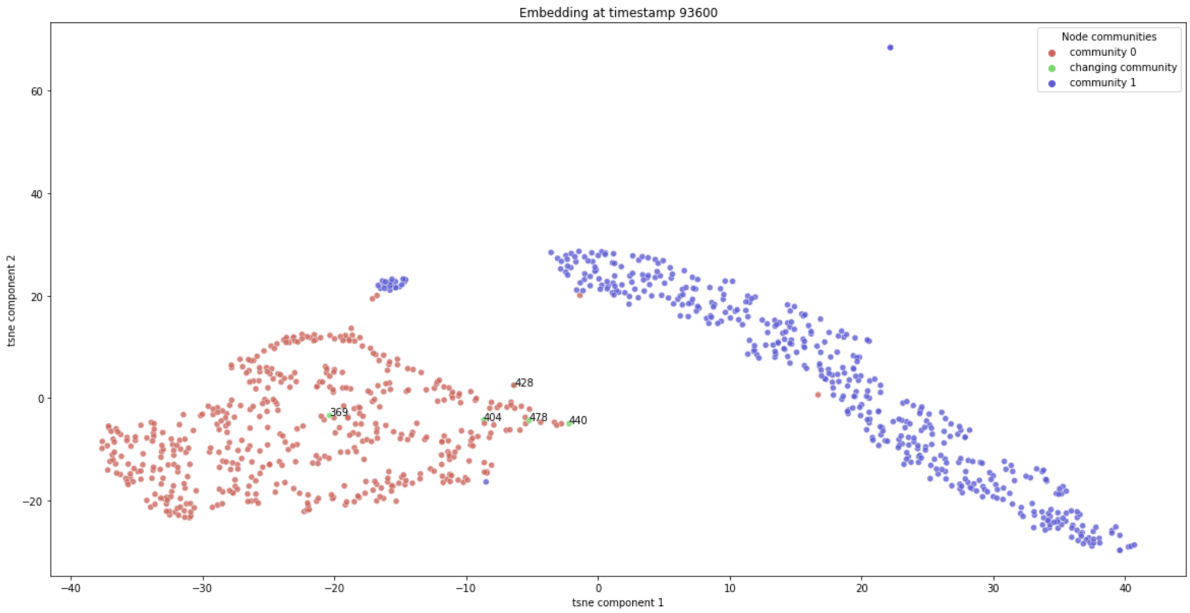

Figure 2 : As the changing nodes are having more interactions with community 1, a group of users from this community is embedded closer to community 0

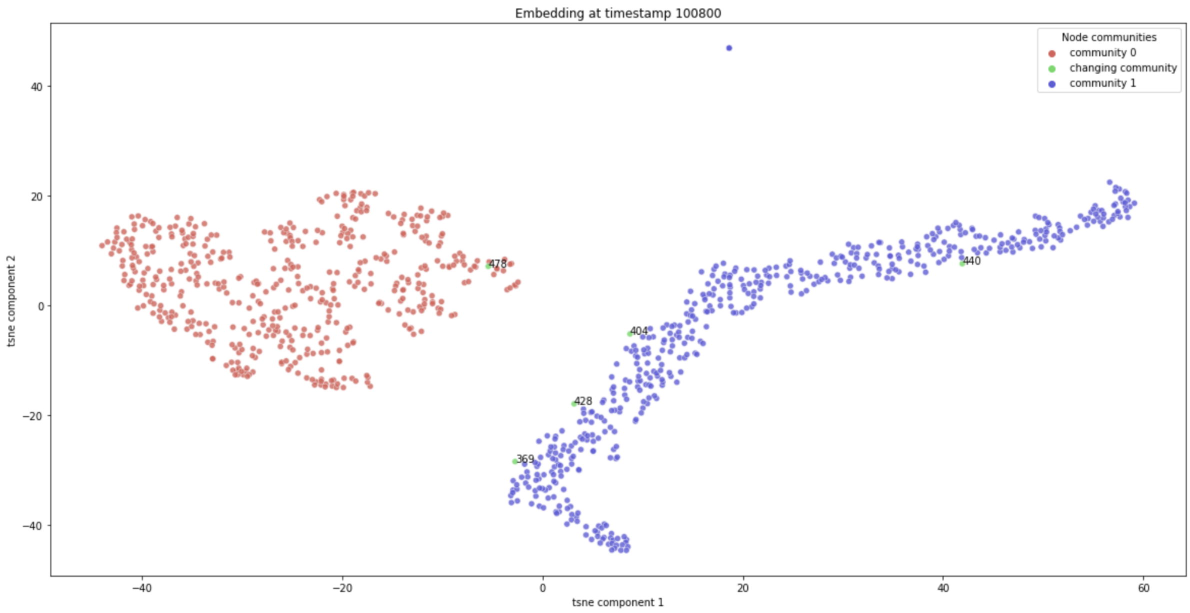

Figure 3 : The model has learned that the changing nodes in green have now a higher probability to interact with community 1 rather than community 0 and thus are embedded close to community 1.

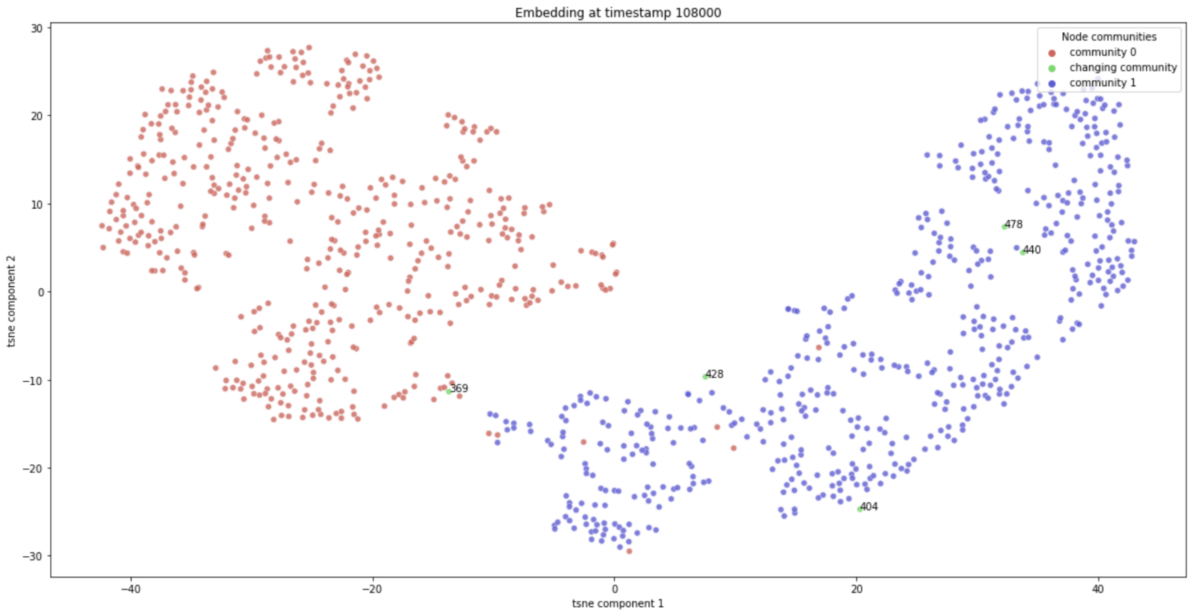

Figure 4 : The nodes that changed community are effectively integrated in community 1 even if they still have far less edges with nodes from this community than with nodes from their original community.

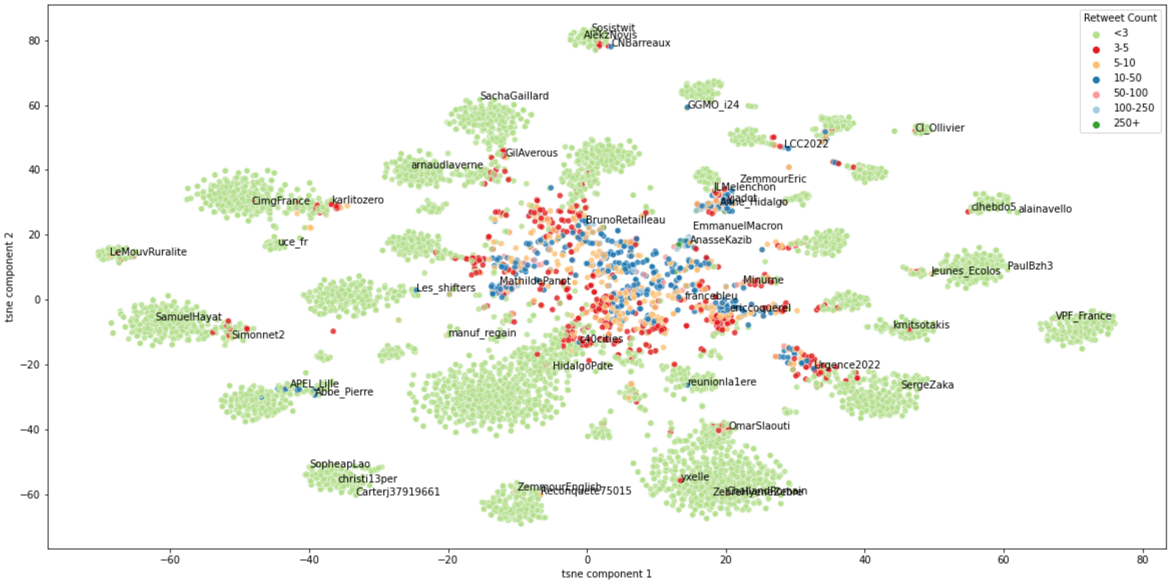

Figure 5 : Embedding of the most retweeted accounts by the candidates of the French election using TGN.
Note. Dataset includes 7 543 twitter accounts and 45 124 retweets. Users are colored by the number of retweets. Projection using t-SNE algorithm.