# Complex and Social Networks
# Assignment 3 - Significance of network metrics

Daniel Benedí García
Irene Simó Muñoz

October, 2023

## Contents



## 1 Introduction

Centrality measures play a pivotal role in network analysis, as they help identify the most influential and well-connected nodes. Closeness centrality, in particular, offers a unique perspective by quantifying how easily information can flow from a node to all other nodes in the network. Its application to syntactic dependency networks (SDN) allows the study of the significance of words in conveying information, their role in sentence comprehension, and their impact on language processing and generation. See a representation of a SDN in Figure 1.

To facilitate the computation of closeness centrality in SDN, this paper introduces the utilization of two distinct models based on two hypohteses.

I The measure of closeness centrality in syntactic dependency networks is significantly modeled by binomial (Erdos-Renyi) graphs, keeping the original numbers of edges and vertices.

II The measure of closeness centrality in syntactic dependency networks is significantly modeled by randomized graphs preserving the original degree sequence. The model depends on the original list of edges and a parameter $Q$ governing the repetitions of the random graph generation.

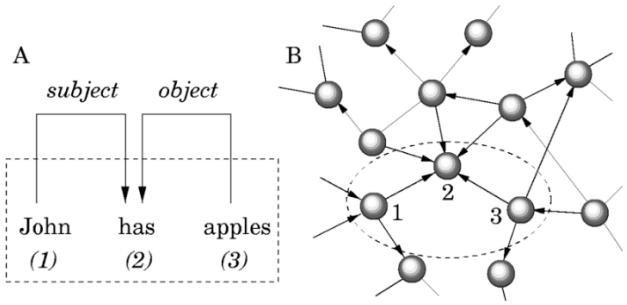



Figure 1: (a) The syntactic structure of a simple sentence. Here words define the nodes in a graph and the binary relations (arcs) represent syntactic dependencies. Here we assume arcs go from modifier to its head. The proper noun "John" and the verb "has" are syntactically dependent in the sentence. John is a modifier of the verb has, which is its head. Similarly, the action of has is modified by its object "apples." (b) Mapping the syntactic dependency structure of the sentence in (a) into a global syntactic dependency network. Extracted from [2]

These models offer a more efficient computational approach to evaluating closeness centrality while preserving the essential structural characteristics of the original networks.

The report studies the significance of the closeness centrality metrics obtained through these models in comparison to the original syntactic dependency networks. We seek to analyze how the metrics derived from the binomial (I) and degree-preserving random graph (II) models compare to those from the original networks. By doing so, we aim to tackle whether these simplified representations offer meaningful insights and faithfully capture the centrality of words in sentence structures.

This document is strucutred in four different sections. Next, results are presented in section 2. Discussion and conclusions are covered in section 3, and finally, some aspects of the methodology are presented in section 4.

## 2 Results

This section presents the results of the significance study of the closeness centrality based on the described hypotheses.

Table 1 shows a summary of our data that we are going to use to test our hypothesis. We can observe that the different SDN have a wide range of number of nodes. There is small networks such as Basque with around 12k nodes and big networks like Czech with almost 70k nodes. All the networks are sparse, the density of edges is low. Despite its sparsity, the mean degrees are quite different with some cases around 4 and other around 13.

In order to study the importance of the network closeness centrality, we want to evaluate if this metric is significantly large with regard our random graph models (our null hypothesis). Therefore, we obtained experimentally which is the probability of the closeness centrality of the random graph to be greater than the input graph, $P(\mathcal{C}_{NH} \geq \mathcal{C}) \approx \frac{1}{T} f(\mathcal{C}_{NH} \geq \mathcal{C}) \leq \alpha$. Table 2 shows that our null hypothesis is correct. Our

| **Language** | **#Nodes** ($N$) | **#Edges** ($E$) | **Mean degree** ($k$) | **Density of edges** ($\delta$) [$10^{-4}$] |
|---|---|---|---|---|
| Arabic | 21532 | 68743 | 6,39 | 2,97 |
| Basque | 12207 | 25541 | 4,19 | 3,42 |
| Catalan | 36865 | 195075 | 10,69 | 2,90 |
| Chinese | 40298 | 180925 | 8,98 | 2,23 |
| Czech | 69303 | 257254 | 7,42 | 1,01 |
| English | 29634 | 193078 | 13,03 | 4,40 |
| Greek | 13283 | 43961 | 6,62 | 4,98 |
| Hungarian | 36126 | 106681 | 5,91 | 1,63 |
| Italian | 14726 | 55954 | 7,60 | 5,16 |
| Turkish | 20409 | 45625 | 4,47 | 2,19 |

Table 1: Summary of the properties of the degree sequences of the studied languages.

networks have a closeness centrality larger than our null hypothesis models.

| **Language** | **Closeness Centrality** | **p-value (Binomial)** | **p-value (Switching)** |
|---|---|---|---|
| Arabic | 0.32597 | 0 | 0 |
| Basque | 0.26981 | 0 | 0 |
| Catalan | 0.34222 | 0 | 0 |
| Chinese | 0.32564 | 0 | 0 |
| Czech | 0.30661 | 0 | 0 |
| English | 0.34381 | 0 | 0 |
| Greek | 0.31403 | 0 | 0 |
| Hungarian | 0.28971 | 0 | 0 |
| Italian | 0.32763 | 0 | 0 |
| Turkish | 0.36075 | 0 | 0 |

Table 2: Summary of the obtained p-values for the experiemnts performed.

Figure 2 shows the obtained results for closeness centrality computation for both models. It is clear that the performance of both studied models is very different; all results for the Binomial model lie further from the measure than those obtained with the Switching model. Other interesting characteristics are revealed by the plot; the variance of the obtained results is not the same for all languages. Observe, for instance, how Basque presents a cloud of points more spread than Catalan, both for the Binomial and Switching models. This probably can be explained by the fact that the number of edges is smaller and therefore the metric is less stable.

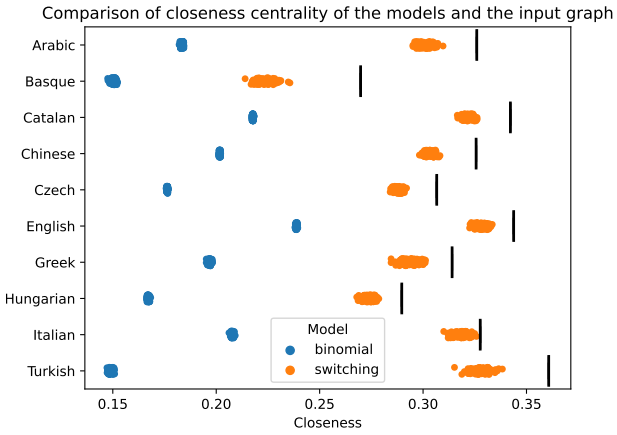



Figure 2: Distribution of the closeness centrality of the different models and the input graph

# 3 Discussion

This section deep dives into the results before presented. The models chosen generate graphs that sim to be hihgh-fidelity reproductions of the original SDN. Both the generation dynamics and the parameters chosen to replicate the SDN structure have an impact on the result.

## 3.1 Binomial model

The results for this model show errors around 35% in the computation of the closeness centrality - see Figure 2-.

## 3.2 Switching model

The results for this model show errors up to 11,5% (for Basque), but are around below 5% in most cases.

The switching model along its implementation is described in detail by algorithm 1 in section 4. The goal of this model is to generate randomized graphs that simulate the original SDN based on the network structure (edges) and a tunable parameter that dictates the number of switchings performed. A switching is the swapping of two edges of the same graph.
The presrevation of the original degree sequence in this dynamics yields two relevant issues natural to the model, briefly discussed in the following paragraphs.

**On the preservation of the original degree sequence** The type of switchings that guarantee the validty of the resulting graph are those preserving of the original degree sequence. However, picking edges uniformly at random does not always result in switchings of this kind.

**On the allowing of forbidden eges** Forcing the preservation of the degree sequence while ensuring randomness can result in a violation of the natural graph structure of the network. Notice that edge switching

by picking the edges u.a.r. can yield non-simple graphs - this is, with multi-edges -, or create loops, which invalidates the graph as a syntactic dependency network since these networks must always be trees [2]. More discussion on the correct performing of switchings can be found in the Methods, section 4.

The results for this model are much more precise than those of the Binomial one, as clearly pictured in Figure 2.

### 3.3 Some relevant questions

**On the null hypothesis** The studied hypothesis were I (Binomial) and II (Switching), as presented in the introduction. For these two models tested, the metric of the closeness centrality is significantly larger for ...

**Language analysis** The set of studied languages is diverse in the sense that contains languages from different linguistic families. In general terms, closeness centrality provides an intuition of how efficient is the network at spreading information; in the case of SDN it indicates how well mixed are the words of the corpus in real sentences of the language. This indirectly provides some sense of other linguistic characteristics; flexibility of sentence structure, morphology (or word inflexion), sentence length, richness of idiomatic constructions or collocations and more.

For example, some languages, like English, have a more linear structure, while others, like Arabic, have a more flexible word order. This can affect how words are connected in the syntactic Languages with rich inflectional systems, such as Czech and Turkish, may have more complex networks due to the different forms a word can take, potentially affecting centrality.

**Bounds on closeness centrality** We comment the results of Hu et al. (2022) [4] on bounds for closeness centrality $\mathcal{C}$. Their work provides plenty of bounds for closeness centrality, but these are often dependant on the mean distance of the graph, which for large scale-free networks can be an even more computationally demanding task than the simple closeness centrality computation. However, some simpler results such as expression 7 of Corollary 2 - expression (1) in this report - or Corollary 8 - expression (2) in this report - can be used to draw some extra information from the results.
The bounds for each studied language following equations (1) and (2) can be seen in Table 3.

$$\mathcal{C} \geq \frac{3}{N+1} \qquad (1) \qquad\qquad \mathcal{C} \leq 1 - \frac{2}{N} + \frac{4(N+1)}{N\left((N-1)(N+2) - 2E\right)} \qquad (2)$$

It is evident that these bounds are weak and do not provide useful information in terms of the actual value of the closeness centrality. However, the lower bounds suggest an ordering of the measures, from whcih we would expect

### 3.4 Conclusions

Some previous works have already covered some results provided in this report. For instance, it has been suggested that "the organization of syntactic networks strongly differs from the Erdös-Rényi graph" [2]. This is confirmed by our results (figure 2), which showed that the closeness centrality is far from the input SDN.

On the other hand, we expected the switching model to have a $p$-value higher than 0 (see appendix A), because it keeps the degree sequence intact, so it was reasonable that the closeness centrality would not be

| **Language** | **#Nodes** ($N$) | **#Edges** ($E$) | **Lower bound** [$10^{-4}$] | **Upper bound** ($\delta$) [$10^{-4}$] |
|---|---|---|---|---|
| Arabic | 21532 | 68743 | 1,39 | 9,99 |
| Basque | 12207 | 25541 | 2,46 | 9,99 |
| Catalan | 36865 | 195075 | 0,81 | 9,99 |
| Chinese | 40298 | 180925 | 0,74 | 9,99 |
| Czech | 69303 | 257254 | 0,43 | 9,99 |
| English | 29634 | 193078 | 1,01 | 9,99 |
| Greek | 13283 | 43961 | 2,26 | 9,99 |
| Hungarian | 36126 | 106681 | 0,83 | 9,99 |
| Italian | 14726 | 55954 | 2,04 | 9,99 |
| Turkish | 20409 | 45625 | 1,47 | 9,99 |

Table 3: Summary of the bounded values of the closeness centrality, using expressions (1) and (2) from [4].

too far from the input graph. But experimentally, we obtained that the $p$-value was 0 (see table 2). In contrast, our supposition that it would not be far from the metric of the input graph was correct as shown in figure 2 because the experimental values were close, although not close enough. We suspected that this was caused by the fact that we approximate the closeness centrality instead of computing it exactly. Therefore, we performed a validation (check section 4.3) which proved us that it was not a matter of our approximation.

We can conclude that both the binomial and the switching model does not keep the centrality closeness neither is greater for any tested language syntactic dependency networks. Although, the switching model does a better approximation.

# 4 Methods

Mean closeness cenrtality $\mathcal{C}$, as presented in [6], is defined as follows:

$$\mathcal{C} = \frac{1}{N}\sum_{i=1}^{N}\mathcal{C}_i \qquad \mathcal{C}_i = \frac{1}{N-1}\sum_{\substack{j=1\\ j\neq i}}^{N}\frac{1}{d_{ij}} \qquad d_{ij} = |P|$$

Where $N$ is the number of nodes, $\mathcal{C}_i$ is the closeness centrality of vertex $i$, $d_{ij}$ is the geodesic distance between vertices $i$ and $j$, also as defined by [6], and $P$ is the shortest path connecting edges $i$ and $j$ in the graph defined as a sequence (then $|P|$ is the length of said path).

## 4.1 Description of the models

**Erdös-Rény model** The Erdös-Rény model, also called binomial model, generates a random graph with a binomial process by flipping a coin and choosing whether an edge should be included or not. The used model is a slight variation of this one, because of instead of choosing the probability of an edge to be included or not, so the number of edges would follow a binomial distribution with expected value $\mathbb{E}[E] = p|E|$, the number of edges is fixed.

**Switching model** The switching model is a generator of random graphs given a degree sequence. This model has an interesting property to our experiments, because it preserve degrees although uniform sampling is not warranted.

**Algorithm 1** Switching mdodel

**Require:** $G = (V, E), Q \in \mathbb{N}$
  **for** $|E| \cdot Q$ times **do**
    $(u, v) \leftarrow$ Choose u.a.r from $E$
    $(s, t) \leftarrow$ Choose u.a.r from $E$
    **if** flip coin gets heads **then**
      Swap edges to $(u, t)$ and $(s, v)$
    **else**
      Swap edges to $(u, s)$ and $(v, t)$
    **end if**
    **if** swap generates self-loops or parallel edges **then**
      Undo swap
    **end if**
  **end for**

The generation of graphs given a degree sequence yields multiple challenges adressed by some pieces in the literature [1, 3, 5]. Our model deals with some of them, but to do so it lacks correctness in other areas. We believe this trade off is beneficial since the flawed properties of our implementation can be bounded. Some heuristics in our code are worth highlighting, referring to the Switching model:

- Unbiased edge-picking.
  Asides from selecting a pair of edges to switch, we also check for possible switching randomly. Checking is referred not to the identification of failure switchings - those invalid because they don't preserve the degree sequence, but valid otherwise - but to the identification of switchings that pick the same edge to switch. Notice that since the graph is undirected, random selection of edges can yield a selected pair of edges $uv$,$vu$, where both $u, v \in V$ that are, in fact, the same edge. If this is not done, there is a bias because in our representation an edge is a pair of numbers in which the first one is smaller than the second. This bias causes that edges between nodes with smaller id or with high id are less probable.

- Guarantee of degree sequence preservation and impact on closeness centrality.
  The described implementation of the Switching model is flawed in the sense that it does not guarantee that no cycles are created. This can yield the generation of non-tree graphs, that although can otherwise show properties similar to those of SDN, fail to be acyclic as SDN are. This has a direct impact on the simulated closeness centrality. More detailed reasoning of this argument can be found in Appendix A.

## 4.2 Implementation and execution

The models have been implemented in C++ using the Boost Graph[1] library [7]. We choose this library because it already implements some useful algorithms that we needed and also provides some representations for the graph. We used a adjacency list to represent the graph because the density is of edges (check table

[1]See `https://www.boost.org/doc/libs/1_82_0/libs/graph/doc/index.html`

1) is low and this representation is really good for sparse graphs. In the switching model, we didn't used this representation because it does not allow to random access to the edges which is a frequent operation needed to this model. Therefore, we opted to use a `std::vector` of pairs of integers to represent the edges and a `std::set` of pairs in order to speed up the edge existence queries.

Compilation flags include those recommended by the Boost library. Moreover, the repetitions of the trials for estimating the $p$-value has been parallelized using OpenMP allowing us to reduce the execution time of our experiments.

The experiments have been run with $Q = 1 + \log E$ for the switching model and the number of repetitions $T = 100$ for all languages. Instead of computing bounds of the closeness centrality or its exact form, we opted to do a Montecarlo approximation with the 10% of the nodes.

### 4.3 Validation of the approximation of the closeness centrality

We chose to do an approximation of the closeness centrality by choosing a subset of the vertices and computing the closeness centrality between them. We decided that using the 10% of the nodes to do this approximation was a reasonable amount in the trade-off between computational cost and approximation to the exact value.

In order to validate that our assumption was correct, we designed an experiment in which we run the method 100 times for the Catalan SDN, a binomial graph with the same number of vertices and edges as the Catalan SDN and applying the switching model $E \log E$ times to the Catalan SDN. Figure 3 shows the results of the experiment in comparison with the exact computation of the metric. We observed that the approximations obtained are pretty close to the exact value, mainly for the binomial graph. The amount of error produced by this method can be observed in the third decimal and is evenly distributed above and below the real value, which allows us to conclude that the approximation is good enough to be used instead of the exact value.

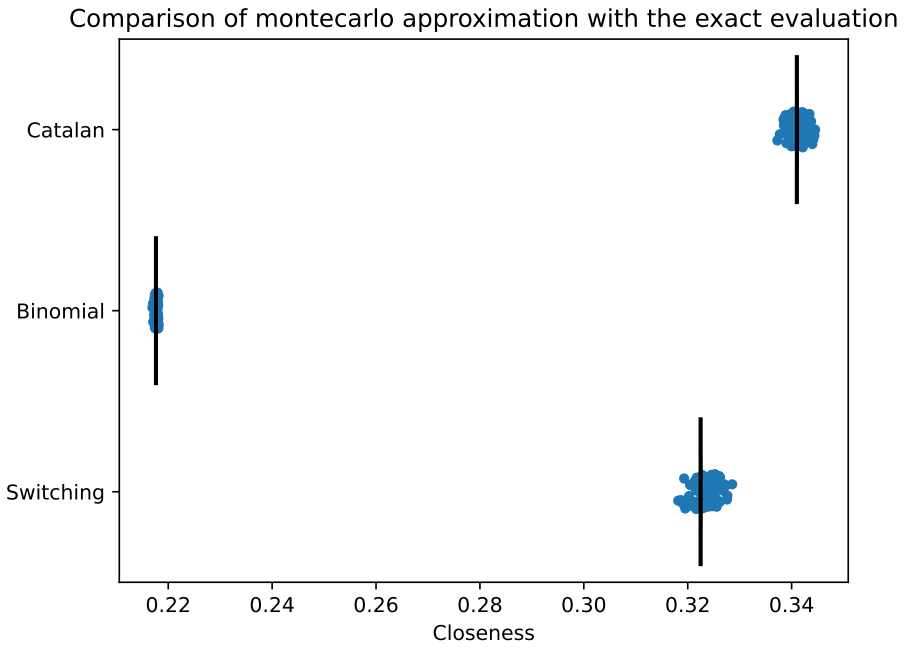



Figure 3: Difference between the approximation and the exact value

# References


[1] Joseph Blitzstein and Persi Diaconis. "A sequential importance sampling algorithm for generating random graphs with prescribed degrees". In: *Internet mathematics* 6.4 (2011), pp. 489–522.

[2] Ramon Ferrer i Cancho, Ricard V Solé, and Reinhard Köhler. "Patterns in syntactic dependency networks". In: *Physical Review E* 69.5 (2004), p. 051915.

[3] Antoon CC Coolen, Andrea De Martino, and Alessia Annibale. "Constrained Markovian dynamics of random graphs". In: *Journal of Statistical Physics* 136 (2009), pp. 1035–1067.

[4] Xin Hu, Abdellah Islam, and Thomas Britz. "Bounds on the closeness centrality of a graph". In: *arXiv preprint arXiv:2204.11283* (2022).

[5] Ron Milo, Nadav Kashtan, Shalev Itzkovitz, Mark EJ Newman, and Uri Alon. "On the uniform generation of random graphs with prescribed degree sequences". In: *arXiv preprint cond-mat/0312028* (2003).

[6] Mark Newman. *Networks*. Oxford university press, 2018.

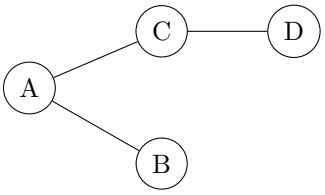
[7] Jeremy G Siek, Lie-Quan Lee, and Andrew Lumsdaine. *The Boost Graph Library: User Guide and Reference Manual, The*. Pearson Education, 2001.

# A Short explanation of why Centrality Closeness should increase with switching

Since syntactic dependency networks are always trees [2], we believe that the switching method should increase the centrality closeness. This is motivated by the fact that a tree does not contain any closed path, but our implementation of switching model does not warranty this property. Therefore, with high probability, it will introduce any closed path. The existence of closed paths provokes that the distances in the graph get reduced, and by the definition of the closeness centrality $\mathcal{C} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{N-1}\sum_{j\neq i}\frac{1}{d_{i,j}}$ it will increase.

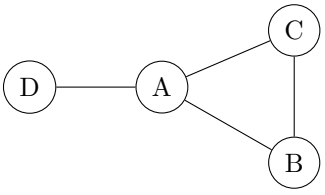
## A.1 Intuition

This fact can be observed in the following example. Given a tree, we can compute its centrality closeness:




$$\mathrm{D} = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 2 & 3 \\ 1 & 2 & 0 & 1 \\ 2 & 3 & 1 & 0 \end{pmatrix} \quad \mathcal{C}_i = \begin{pmatrix} 5/6 \\ 11/18 \\ 5/6 \\ 11/18 \end{pmatrix} \quad \mathcal{C} = \frac{13}{18}$$

Assuming that our switching model selects the edges $(A, B)$ and $(C, D)$ to switch into $(A, D), (B, C)$. Then we get the following graph and centrality closeness:




$$\mathrm{D} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 2 \\ 1 & 1 & 0 & 2 \\ 1 & 2 & 2 & 0 \end{pmatrix} \quad \mathcal{C}_i = \begin{pmatrix} 1 \\ 5/6 \\ 5/6 \\ 2/3 \end{pmatrix} \quad \mathcal{C} = \frac{5}{6}$$

It is clear that in this example, the switching model does not keep the tree property by introducing a closed path which has reduced the distance between vertices and therefore has increased the closeness centrality.

## A.2 Formal proof

A tree is a graph with $n$ nodes and $n-1$ edges. Since the Switching mdoel only swaps already existing edges, both the number of nodes $n$ and edges $n-1$ is fixed. We w.t.s. that the probability that we still have a tree after a number of iterations tends to 0 as the number of iterations tends o $\infty$.

By the definition of our edge swapping, we define the probability of creating a cycle after one iteration where we start with a tree graph as the probability of choosing two edges u.a.r that share a vertex.

- The number of ways to choose one vertex from $n$ vertices is $n$.

- Once we have a fixed vertex, the number of ways to choose two edges that connect to that vertex is C(degree of the chosen vertex, 2). In a tree, the degree of any vertex is at most $n-1$ (it can have at most $n-1$ edges connected to it, that would be a star graph).

- We perform the switching depending on the outcome of flipping a coin

$$p(n)_T = \binom{n}{1}\binom{n-1}{2}\frac{1}{2} = n(n-1)(n-2)\frac{1}{2} = \mathcal{O}(n^3)$$

Then, the probability of having an edge increases both by repetitions and as $n$ grows.
Containing a cycle increases the degree of a vertex, and vertex with degree 0 are not considered for the closeness centrality computation. Therefore, containing a cycle introduces a positive drift to the closeness centrality measure, we expect to obtain higher values.