Dream Content Analysis with Topic Modeling

CSCI 390 Senior Project & Seminar

Gavin Osborn | 12/13/23

Introduction

Dream content analysis has been done manually by psychologists for some time. Many systems have been developed to quantify and relate the "content" of a given dream, from many different perspectives; notably, the Hall/Van de Castle System[1] (HVC) of quantitative dream content analysis. This system differs from others in being composed from formally defined quantitative accounts of a handful of qualitative categories. Typical content analysis defines content as being the whole of all constituent elements; thoughts, emotions, images, and so on; this is largely empirical in nature when not given the constraints of a formally defined quantitative system, an issue the HVC system avoids.

Each element in a dream is a formally defined code, implemented and "coded" manually by a psychologist going through the dream. The Dream Bank[2] is collection of both coded and uncoded dreams and some manner of statistical analysis on each. The definitions, however formal, are principally judged by a human element capable of error and subject to the reality of lexical ambiguity. My system changes this state of affairs by allowing for a necessarily rule-based system within its own language; that is, it is an unsupervised computational approach to identifying self-defined elements related by a quantitative metric. This largely removes the human element in all but interpreting the final result (as one would interpret a coded dream content system) and circumnavigates the complications arising from the recombination of different dream elements when certain elements are judged to be more relevant; for instance, the HVC system allows for a combinatoric explosion in the number of descriptors for a dream by allowing for a recombination of different dream elements together into new variables[3]. Though a formal definition, this is not something complete in itself by definition without external modification prior to interpretation.

Furthermore, because this system is composed of only mathematically derived elements and relations, there is little room for personal or cultural bias, and the system works for all languages with proper preprocessing. I believe the system to have great potential for use as a therapeutic tool, allowing for the discovery of connections otherwise not visible to an individual when attempting to interpret the meaning behind their dreams or the reason behind recurrent patterns they believe to be evident within them. With proper systems of inductive inference in place, the system may be able to be expanded to further discover implicit connections and predict future dream patterns; and this may also be able to have use beyond just dream interpretation.

1 https://dreams.ucsc.edu/Coding/
2 https://www.dreambank.net/
3 https://dreams.ucsc.edu/Info/content_analysis.html

Text Processing

For the purpose of testing and experimentation, I used the Dream Bank dataset, saved as a single xml file. I treated each dream journal as its own corpus. The corpus was tokenized, contractions were expanded, and html tags were removed alongside stop words.

These tokens are used to create a TF-IDF values matrix, where each vector is effectively a word embedding representing word occurrence by document, weighted by relative significance within each document. These values can be used to fit dimensionality reduction models.

Topic Modeling

A matrix that relates one thing to another may suffer from high dimensionality that provides more individual information than needed, complicating any analysis of the data, both computationally and qualitatively. Different techniques exist to reduce dimensionality, topic modeling being a class of such techniques that clusters the data into “topics”. I’ve used Non-Negative Matrix Factorization (NMF) for the purpose of reducing the dimensionality of the co-occurence matrix for individual dream journals. There were many other algorithms to choose from that could do the same thing, but NMF is considered to be the best in providing topic relationships[4]. Considering what the algorithm allows given its simplicity, NMF was the best choice for the project.

NMF factorizes the given matrix (V) into two component matrices W and H, which when taken as a dot product, provide an approximation of the original matrix, V. A more formal description of the algorithm with a practical implementation exists,[5] but the gist of it is that the minimum distance between the approximate and original matrices are found using a process that iteratively adjusts the values by a ruleset determined from the calculation of a gradient between any two values between the two matrices.

One matrix will take the rows and the other the columns, and both will reduce the dimensionality by way of clustering the vectors by a predefined number of topics; this is the primary drawback of using NMF over something like Latent Dirchlet Allocation. In our case, the W matrix is a document x topic matrix, and the H matrix is topic x term.

4 https://www.frontiersin.org/articles/10.3389/fsoc.2022.886498/full

5 http://www.quuxlabs.com/blog/2010/09/matrix-factorization-a-simple-tutorial-and-implementation-in-python/

Network Pruning

Each component matrix of the NMF model may be graphed in two ways, as they are essentially matrices representing two relations; the topics by some metric, and whatever the topics are contrasted against by their topic relations. Thus there are four ways to graph the model. The cosine similarity (normalized dot product, each value effectively being a percentage of weight) of each chosen vector may be taken to get such relations as a graphable symmetric matrix. When graphed, this provides a completely connected graph and does not necessarily show relevant connections; some metric must be employed to reduce the number of edges within the graph.

There are many ways to do this, and innovation within these areas is a central concern of proximity data and semantic analysis. I focused on Pathfinder Associative Networks, and also produced some graphs with a Nearest Neighbors approach. In any case, it is possible to decide upon a metric for salience and prune the graph with such information in mind. Furthermore, because they provide a method for pruning connections, they can also identify indirect relationships between elements in the graph by their salience metric; the graph itself becomes representative of relationships that were not there nor visible prior to pruning.

The Pathfinder algorithm prunes connections with distances exceeding that of alternate paths; paths violating the triangle inequality. It does so by first ordering all connections by a non-decreasing order and clustering edges into node sublists[6]. Then the sublists are iteratively considered for their number of eliminated edges, indicating their salient relation. The distance in sum values considered for such pruning is determined by the “r” parameter entitled the Minkowski distance. Typically, this is set to infinity. The maximum number of edges in the alternative paths considered is determined by the “q’ parameter, which must be less than n-1 for all variations of the Pathfinder algorithm, and n-1 provides the sparsest graph that satisfies all properties of a minimum spanning tree.

6 This has the peculiar effect of essentially providing the Pathfinder network with all information available in Hierarchical Clustering. Pathfinder Associative Networks: Studies in Knowledge Organization (1990).

Pipeline

Currently, I have a set of Python scripts collecting data from a single dream journal. I preprocess the text, put it through NMF, then prune the data. I save the labeled Pathfinder Networks (PFNETs) alongside the NMF matrices as .dot and .txt files respectively. The former can be visualized by some sort of graphing tool like Gephi, and the latter can have some metadata appended to them and then used as data files for the official Pathfinder toolset.[7]

7 https://research-collective.com/PFWeb/Download.html

Results

I collected results from two different sets of parameters. The first set of graphs (Set A) were derived from an NMF model seeking 100 topics from 400 words and 421 dream journal entries. The second set (set B) were derived from 10 topics, 100 words, and 421 dream journal entries.

Words by topic relationship has been the central focus of the project, as I believe the word relationships to collectively represent semantic relationships, and thus also represent wider “elements” within the dream. I have included such relationship graphs for both graph sets, but for other categories of graphs I have included only topic by word, and only for set A. If one desires more, this document should have been bundled with the .dot files necessary for visualization.

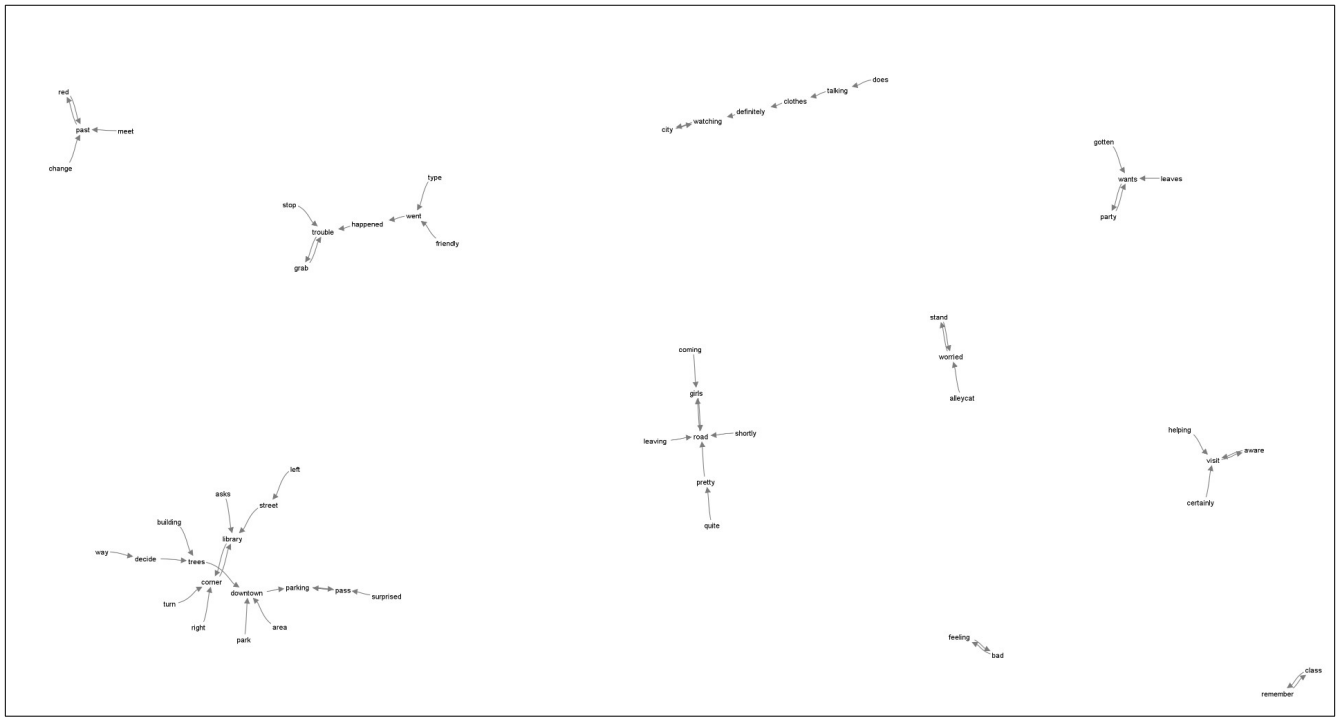

Fig 1. Words related by topic pruned by Nearest Neighbors. A small subset of a 400 word and 100 topic graph.

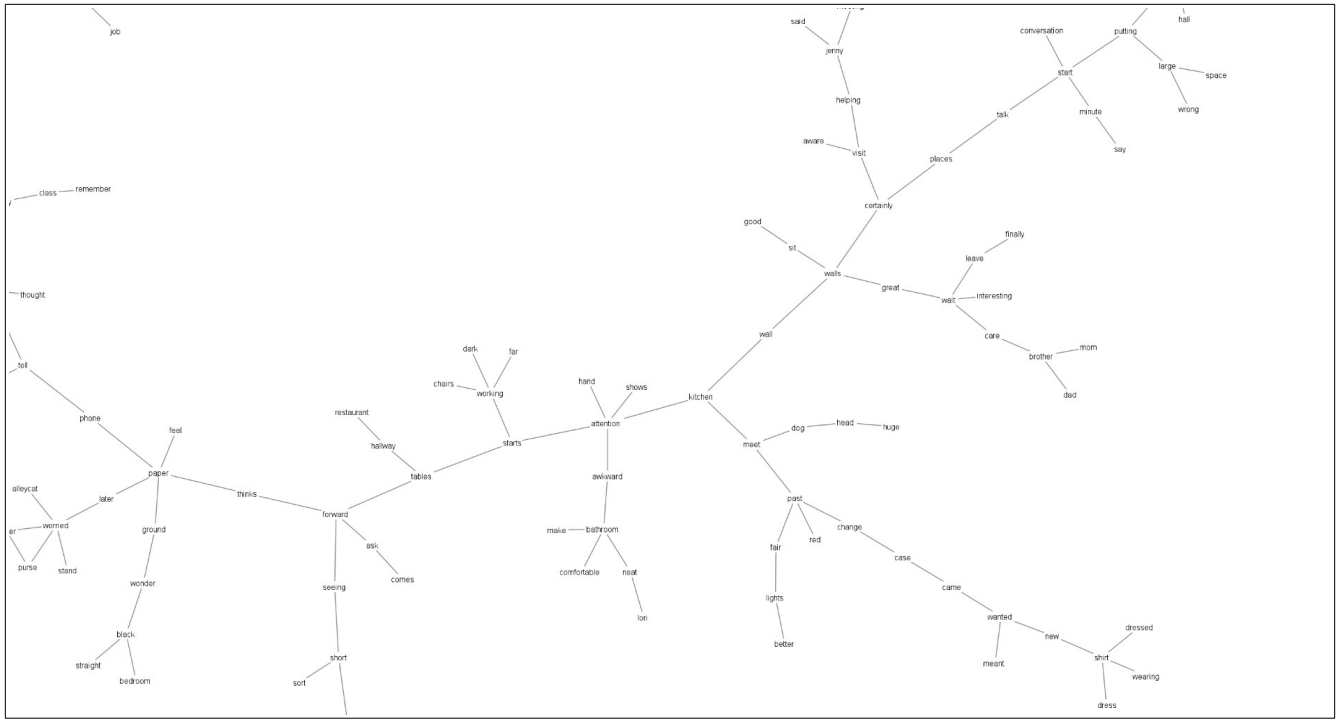

Fig 2. Words related by topic pruned by Pathfinder. A small subset of a 400 word and 100 topic graph.

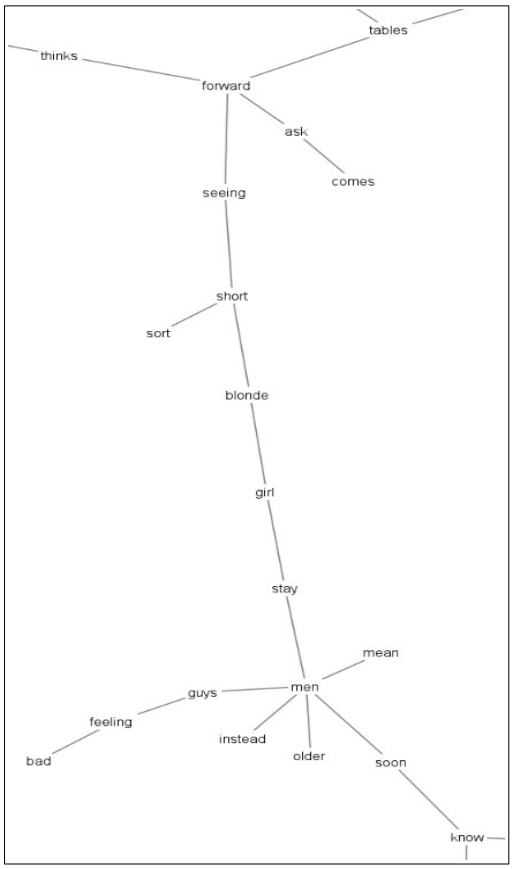

Fig 3. An interesting subtree of words related by topic pruned by Pathfinder. A small subset of 400 words and 100 topics. The element of “older men” associated with negative terms.

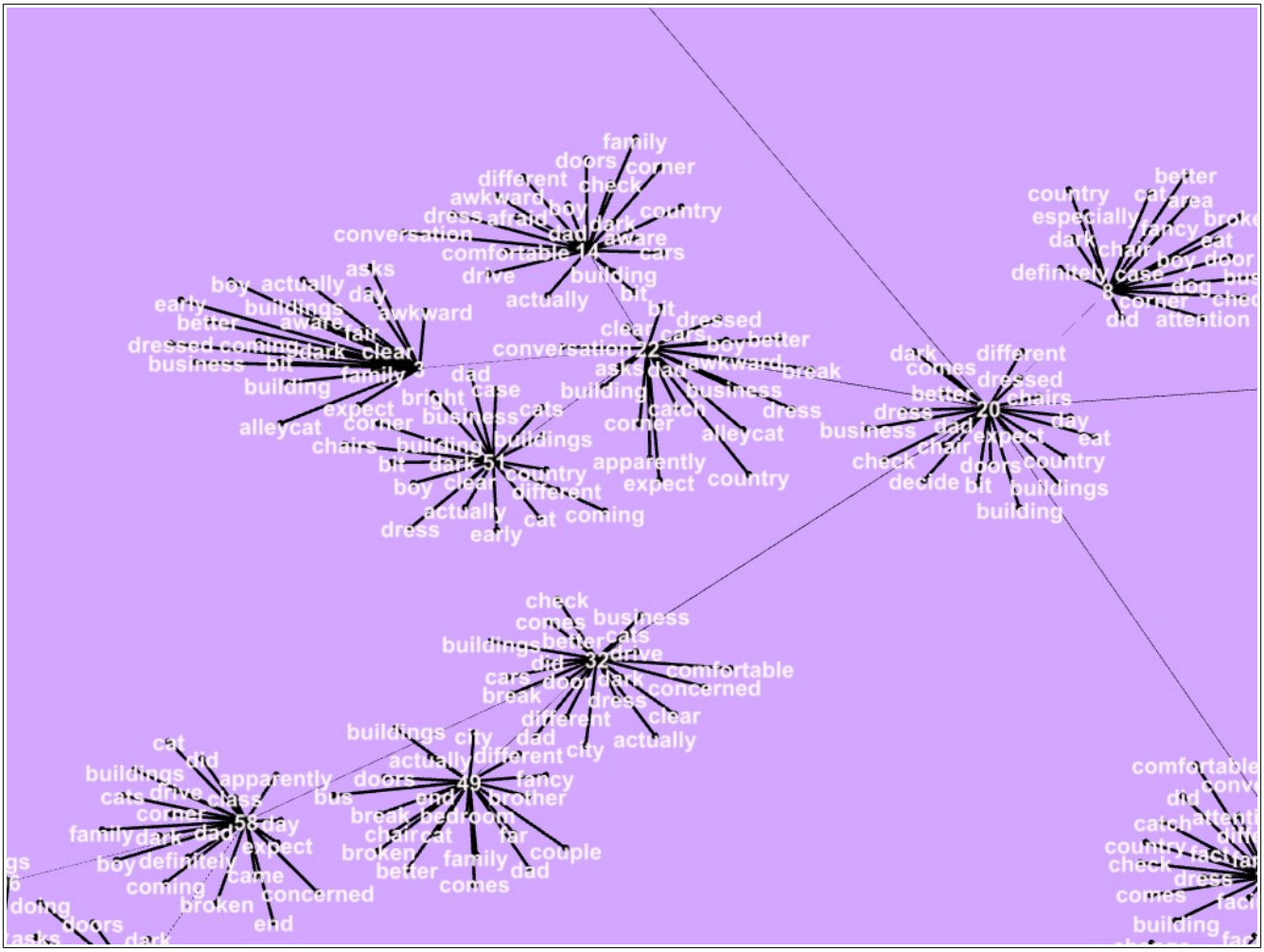

Fig 4. Topics related by word pruned by Pathfinder. A small subset of 400 words and 100 topics. Visualized by Gephi. Very similar topics clustered together. Seem to indicate overarching dream themes, implying that words are related by theme.

These graphs are very large, and when visualized by Gephi, the meta relationships are more visible.

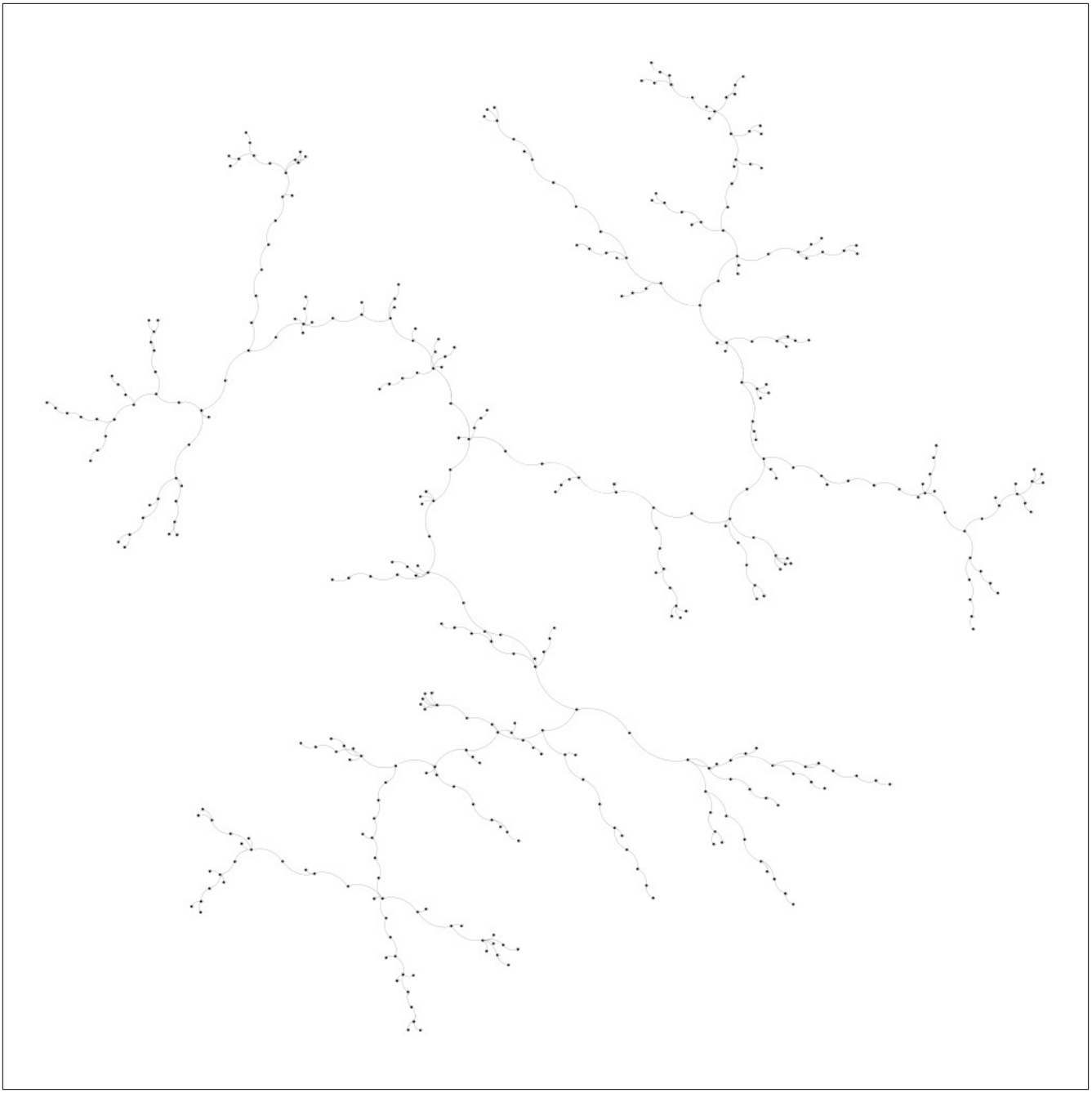

Fig 5. Words related by topic pruned by Pathfinder. 400 words and 100 topics. Visualized by Gephi.

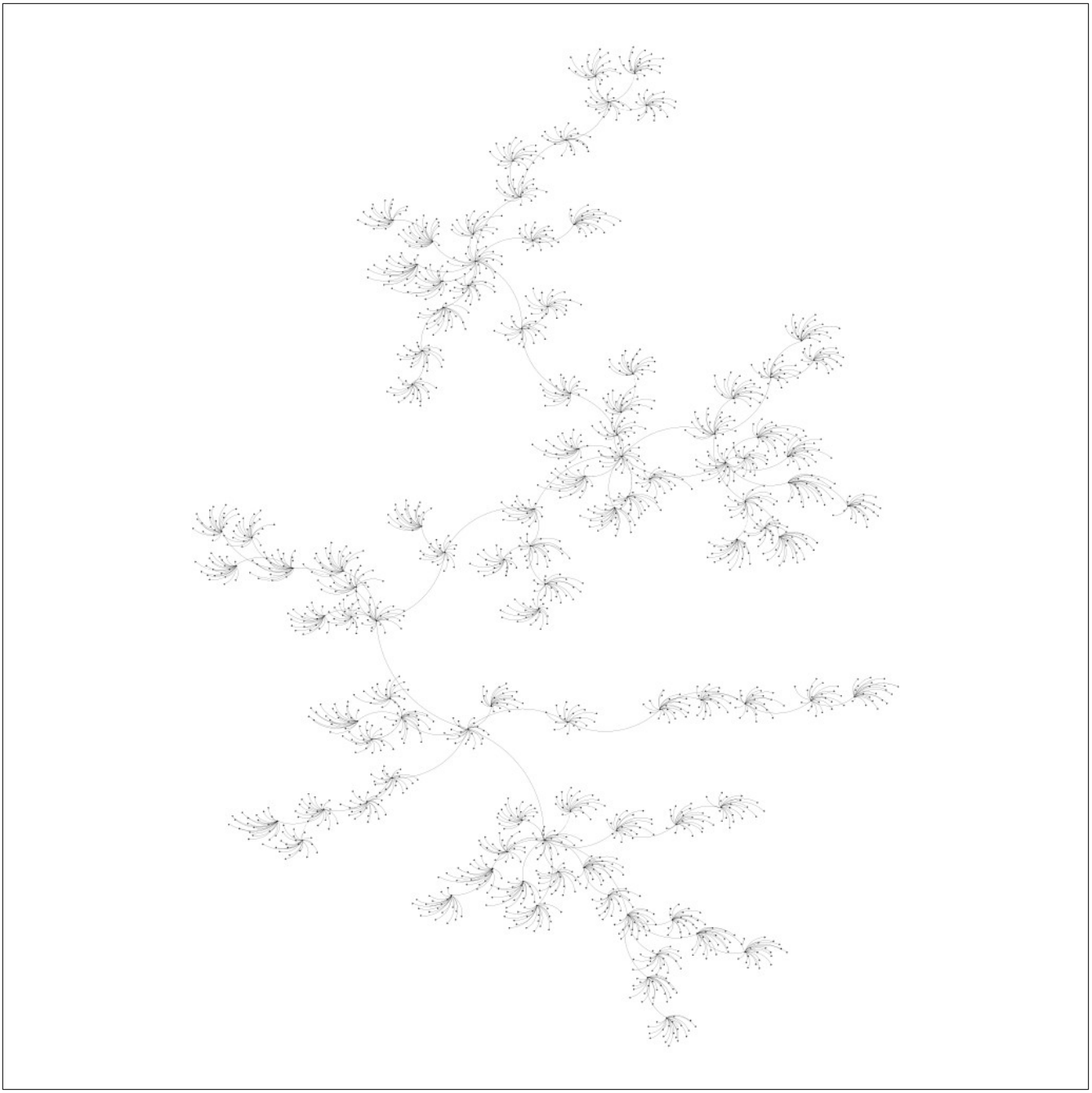

Fig 6. Topics related by word pruned by Pathfinder. A small subset of 400 words and 100 topics. Visualized by Gephi.

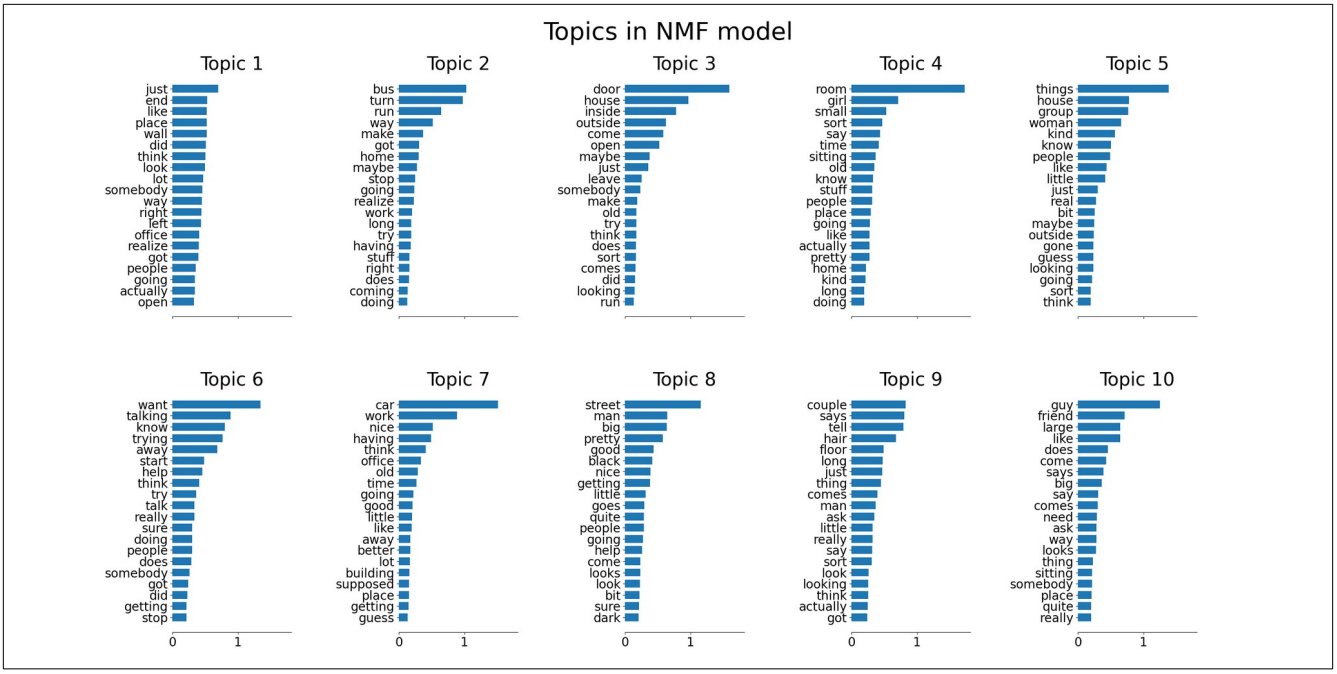

Fig 7. Topics and their top 20 words. Matplotlib visualization of NMF with 100 words and 10 topics.

Potential Applications

The bedrock of this project is in the fundamental principle of distributional semantics; we may infer semantic meaning from usage and context. This project was only able to touch upon the first matter, as the NMF model derives topics only from co-occurrence rather than direct syntactic structure. However, because we can infer semantic meaning by grouping words together, we are able to identify clustered elements within the data and thus also the most salient information within the data. One example of use here is in figure 3, where it appears that older men are associated with negative terms for this particular dreamer. These clustering techniques can be applied universally to all symmetric matrices; in this case, the symmetric co-occurrence based cosine matrix. So too can any symmetric matrix be applied to Pathfinder, whether it be a distance, dissimilarity, similarity, or other metric relating terms. This gives these methods unbelievably broad application, with known use in bioinformatics, citation analysis, and otherwise.

An interesting application of Pathfinder networks and other network pruning algorithms is in inductive inference. On its own with any symmetric matrix, Pathfinder may be able to find implicit connections in the data by preserving only the most salient links. However, doubled upon a statistical model for semantic relatedness that derives indirect connections, such implicit information may become even more visible. This is the motive behind Reflective Random Indexing[8] and Latent Semantic Analysis. Further comparison to NMF is necessary to know exactly which model may be most suitable.

Once able to reliably make inferences between different elements in a salient network, we can do many things. Applications to the digital humanities are abundant; premises may be identified through clustering, if extended to clustering clauses and accounting for parts of speech, and extrapolations or criticism of authors could be expanded. Furthermore, an original application of Pathfinder was in citation analysis; predicting trends and connecting authors to other authors by shared references. This indicates considerable use in trend prediction and recommendation systems. Such systems can thus also lead us to imagine building mental models for artificial intelligence, in various systems; one use case may be as simple as a videogame where a player's actions build up a salient network with possible indirect relations suggesting the NPC to prompt an LLM with information that may allow it to predict a player's future behavior, or speak about events that have yet to occur. This would allow for lifelike behavior.

As for the future of dream content analysis; in conjunction with sentiment analysis, a new coding system for dream content analysis may be developed, or the HVC system may be otherwise expanded. This would allow for personalized coding with a mathematically derived method, and more expansive and automated dream coding could be achieved. The opportunities afforded by semantic analysis are truly very broad, and this project could be expanded in many directions.

8 RRI is considered the best for this. See Reflective Random Indexing and indirect inference: A scalable method for discovery of implicit connections. Cohen et al. (2009).

Projected Timeline

Though the project can be expanded in many ways, and each component can be used elsewhere, there are a handful of things that I may improve upon or do in the near future. I hope to create a UI that allows for an individual to import their own dreams, display their journal and see a list of top words and related documents, and write new journals; I've wanted such a program for quite some time. I have all the fundamental puzzle pieces to do this already. Once this is complete, further work can be conducted on trying out different models that may be pruned by Pathfinder, and then different methods may be employed to standardize the PFNETs for objective measurement. Different parameters can be chosen and experimented with, and we can limit ourselves to different constraints such as only including proper nouns. Properly constrained PFNETs of dreams may be able to be statistically analyzed to compare the dream content of different demographics – or provide further therapeutic use by showing differences between life stages of an individual.

Conclusion

Dream content analysis is an interesting field, and the pipeline that I have developed provides an interesting algorithmic measure to identify dream content. The use cases of these methods are broad and have plenty of room for improvement by experimenting with different combinations of semantic relatedness model, derivations of data for such models, and pruning methods. Other methods of information extraction may be used, such as sentiment analysis or conjoined use with text analysis based upon neural networks. As the project stands, it is a great tool for therapeutic use and as an aid in dream interpretation.