# Graph convolutional networks for exploring authorship hypotheses

**Tom Lippincott**
Johns Hopkins University / Baltimore, MD
tom@cs.jhu.edu

## Abstract

This work considers a task from traditional literary criticism: annotating a structured, composite document with information about its sources. We take the Documentary Hypothesis, a prominent theory regarding the composition of the first five books of the Hebrew bible, extract stylistic features designed to avoid bias or overfitting, and train several classification models. Our main result is that the recently-introduced graph convolutional network architecture outperforms structurally-uninformed models. We also find that including information about the granularity of text spans is a crucial ingredient when employing hidden layers, in contrast to simple logistic regression. We perform error analysis at several levels, noting how some characteristic limitations of the models and simple features lead to misclassifications, and conclude with an overview of future work.

## 1 Background

In this paper, we consider the Documentary Hypothesis (DH),which proposes a specific combination of sources underlying the existing form of the first five books of the Hebrew Bible known as the *Torah* (Friedman, 1987).[1] Table 1 lists the eight sources in the DH and short description. We use "sources" in a more general sense than in straightforward author attribution literature: the labels may resolve to original material from particular authors, but could also be insertions from contemporary sources, redaction by a new liturgical community, translation of another document, and so forth.

Related areas such as authorship attribution and plagiarism detection, that rely on characterizing documents according to *style*, have a long history in the NLP research community (Potthast et al., 2017; Stamatatos, 2009; Potthast et al., 2010) as a text classification (Sari et al., 2018) or clustering/outlier detection (Seidman and Koppel, 2017; Lippincott, 2009) task. They typically consider the situation where the data are isolated document-label pairs without inter- or intra-document structure (Stamatatos, 2009; Seroussi et al., 2011). In contrast, the DH labels are embedded in the book-chapter-verse structure of the Torah. The basic premise remains the same: the labeled texts should contain linguistic features that, in some fashion, reflect their source. Our intuition is that structural information, which is often isomorphic to other modalities (narrative, time of composition, rhetorical role, etc) is a useful signal that can be exploited by a suitable model. For example, one source might tend to make word-level edits distributed evenly across a document, another might insert narrative elements constituting entire chapters, while a third might make ideologically-motivated changes only to the work of an earlier source. These observations all require some awareness of position inside a larger structure, in

| Name | Time period and location |
|---|---|
| **E**lohist | 9th to 7th century, Israel |
| **J**ehovist | 9th to 7th century, Judah |
| **P**riestly | 6th and 5th centuries |
| **1D**euteronomist | 7th century (pre-exilic) |
| **2D**euteronomist | 6th century (post-exilic) |
| **R**edactor | Post-exilic |
| **nD**euteronomist | Single large span in Deuteronomy |
| **O**ther | Assorted (poems, repetitions) |

Table 1: Standard sources for the Documentary Hypothesis of Torah authorship

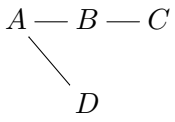
[1] The DH has 150 years of history, exists in several forms, and is by no means universally accepted: for the purposes of this study, it is a reasonable starting point.



*Proc. of the 3rd Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pp. 76–81
Minneapolis, MN, USA, June 7, 2019. 

addition to the linguistic features.

Linguistic features for determining a document's source are often designed for robustness and generalization, e.g. word length, puctuation, function words (Mosteller and Wallace, 1963; Sundararajan and Woodard, 2018). Some studies employ full vocabulary or character n-gram features (Sari et al., 2018), which increase the potential for overfitting on topic and open-class vocabulary, but can also capture additional stylistic aspects. Recent work has begun to apply neural models to the author attribution task: Sari et al. (2018), for example, combine character n-gram embeddings with a single hidden layer feed-forward network. These features and models do not take into account document structure.




| | A | B | C | D |
|---|---|---|---|---|
| A | 1 | 1 | 0 | 1 |
| B | 1 | 1 | 1 | 0 |
| C | 0 | 1 | 1 | 0 |
| D | 1 | 0 | 0 | 1 |

Figure 1: In a GCN, each layer receives input from the previous according to the node adjacency matrix. Initially, node *C*'s representation is based only on it's own features. After the first convolutional layer, it is also based on features from its predecessor *B*. By the third layer, it has access to information propagated from its two-hop ancestor *A*.

The recently-introduced *graph convolutional network* (GCN) (Kipf and Welling, 2016) allows nodes, with $L$ layers of convolution, access to representations of their neighbors up to $L$ hops away. This is accomplished by using a function of the adjacency matrix $A' = f(A)$, which describes the connections between nodes, to determine how the representations from one layer feed into the next. Figure 1 shows a four-node graph and its associated adjacency matrix, plus self-connections (the diagonal) so that nodes employ their own features. Each layer $n$ in the corresponding GCN has a $4xH_n$ output, where $H_n$ is the size of that layer's representations. Before passing the output of layer $n$ to layer $n+1$, it is multiplied by $A'$, which for suitable functions (e.g. $f = norm$) effectively mixes the output for a given node with that of its neighbors. Thus, at layer $l$, each node's representation has been combined to some degree with it's $l$-size neighborhood.

## 2 Experimental setup

Our goal is to train a model to recover the DH using stylistic features: the following sections describe our data, features, and models.

### Data

Our experiments use the Westminster Leningrad Codex (WLC) (Lowery, 2016), available at `http://tanach.us/Tanach.zip`, a publicly-available TEI document (editors, 2019) of the oldest complete Masoretic text of the Hebrew Bible. The WLC encodes the DH as described in Friedman (2003), mapping spans (fragments of the Torah document tree) to sources. Spans can be at different levels of granularity, from book down to token, e.g "Num:20:1.1-Num:20:1.5" or "Lev:23:44-Lev:26:38". Each span corresponds to one or more consecutive nodes in the WLC tree and their children. There are 378 spans with associated source labels, covering the entire Torah. The Torah portion of the WLC consists of 5 books, split into 929 chapters, 5,853 verses, and 79,915 tokens. Furthermore, tokens are segmented into morphs (stems, prefixes, and suffixes), with 6,625 unique morphs averaging 1.5 per token. Our most significant data preprocessing is the removal of vowel pointing, which was not introduced until the middle of the first millenium A.D., at earliest. The WLC is tree-structured, and any location can be specified with a tuple of $(book, chapter, verse, token, morph)$, where the latter two are indices calculated from the data. In this paper we construct our features from morphs, not tokens, as most Hebrew function-words occur at the prefix/suffix level.

The data points are the labeled spans of the DH: the categorical source value, and some linguistic or structural features extracted from the corresponding fragment of the WLC. As recognized by much previous work (Mosteller and Wallace, 1963), authors can often be trivially distinguished using naive vocabulary features, and care must be taken to avoid this uninformative result. We therefore construct bag-of-morph distributions limited to those morphs that occur in every source, as a simple heuristic to focus on the distribution of function-words and widely-used open class vocabulary. This reduces the morph vocabulary from 6,625 to 70. On inspection, these appear to be ~50% function-morphs, ~20% verbs, ~20% common nouns, and three proper names: Moses, Is-

rael, and Jehovah.

We also consider two structural features: first, indicator variables for the span's level of **granularity** (books, chapters, verses, or words), with the idea that sources differ in the processes that inserted them, e.g. broad original narratives versus surgical edits. Second, and separate from the feature vectors, we construct a **sibling** adjacency matrix for the spans, where a span is connected to another if they share the same parent in the WLC (e.g. if the span is a sequence of chapters in Genesis, the parent is the Genesis book node). This will allow graph-aware models to consider how a source is situated relative to nearby sources.

### Models[2]

Our baseline models are logistic regression (LR), a standard non-neural classification model capable of handling heterogeneous and potentially-correlated features, and multi-layer perceptrons (MLP), the structure-unaware corrolary to the simple GCN architecture we employ:

**LR** Logistic regression is equivalent to a neural network with a single fully-connected linear mapping feature vector to label distribution

**MLP** A multi-layer perceptron maps the input feature vector through $L$ fully-connected hidden layers of dimensionality $d_1, d_2 \dots d_L$, each followed by an activation function

**GCN** Graph convolutional networks (Kipf and Welling, 2016) are similar to MLPs, but at each hidden layer the current *matrix* containing hidden states of *all* data points is multiplied by the adjacency matrix, allowing a data point to take its neighbors' states into account

The final layer (or, in the case of **LR**, the input) is fed to a fully-connected linear layer that projects it to the number of labels, followed by softmax to get a valid distribution. For **MLP** and **GCN**, We experiment with linear and non-linear (ReLU) activations, with 32-unit hidden representations based on dev set grid search over possible sizes in $(16, 32, 64, 128)$. All models can be trained with or without the granularity indicator variables (**gran**). The **GCN** models are also passed the sibling adjacency matrix: combined with one hidden layer, this allows the models to take into account properties of adjacent spans.

The labeled spans are randomly split into 80/10/10 train/dev/test. Because the data set is very small, we can treat it as a single large batch, which also simplifies the GCN approach, and train by only back-propagating error from the training set loss. We use the Adam optimizer with default parameters ( $lr = 0.001, betas = (0.9, 0.999)$ ) and allow up to 10k epochs, and monitor the dev set loss for early stopping after 100 epochs without improvement. We report macro F-scores on the test set, which gives equal weight to the eight source labels.

## 3 Results

Table 2 shows the performance of the model and feature combinations described in Section 2. Our primary result is that **GCN**, with ReLU activation and the granularity features, outperforms the other configurations. Perhaps most striking is the importance of the granularity features for the models with hidden layers. While these indicator variables hurt performance of logistic regression, the rest of the models all see ~10-20 point improvements. Interestingly, when using the full feature set (i.e. allowing the model to consider topic), including granularity features dramatically and consistently *lowers* performance: with only word features, all GCN and MLP models manage an F-score ~77, but with the granularity indicators this drops to ~56. The granularity features may allow for particularly damaging overfitting, and we plan to explore this in follow-up work.

| Model | F-score |
|---|---|
| LogisticRegression | 45.80 |
| LogisticRegression+gran | 41.39 |
| GCNstruct+lin | 11.24 |
| GCNstruct+relu | 7.92 |
| MLP+lin | 27.79 |
| MLP+lin+gran | 45.22 |
| MLP+relu | 24.97 |
| MLP+relu+gran | 47.45 |
| GCN+lin | 31.38 |
| GCN+lin+gran | 46.64 |
| GCN+relu | 28.77 |
| GCN+relu+gran | **48.60** |

Table 2: Performance of different model and feature configurations on the test set, in terms of macro F-score

[2]Code available at www.github.com/FirstAuthor/documentary-hypothesis

Table 3 shows the confusion matrix of the best model (GCN+relu+gran). The P source is more than twice as likely to be misclassified as J than as E, perhaps reflecting their shared provenance in Judah and concern with the Aaronic priesthood. The P and R sources also show affinity, again, with the latter thought to have arisen in Judah (or Babylon) long after Israel ceased to exist.

| Gold | Guess | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | J | E | P | 1D | 2D | nD | R | O |
| J | 100 | 8 | 7 | 0 | 0 | 0 | 3 | 0 |
| E | 22 | 53 | 8 | 0 | 0 | 0 | 0 | 0 |
| P | 13 | 5 | 77 | 0 | 1 | 0 | 4 | 0 |
| 1D | 2 | 0 | 2 | 7 | 1 | 0 | 0 | 0 |
| 2D | 2 | 2 | 1 | 0 | 5 | 0 | 0 | 0 |
| nD | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R | 3 | 3 | 11 | 0 | 0 | 0 | 33 | 0 |
| O | 2 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |

Table 3: Confusion matrix of the eight labels for GCN+relu+gran, where entry $(r, c)$ is the number of times label $r$ was misclassified as label $c$

Table 4 lists the ten most-misclassified spans, based on the difference between the probability of the guessed label and the correct label. Looking closely at a few misclassified spans, we make some (amateur) observations: the P and J sources share an affinity for the word "wife",[3] sometimes inserting a clarification of the E source that otherwise paints a less-than-monogamous picture. However, combined with our bag-of-words assumption this can create problems: Genesis:25:1-4 is labeled E but misclassified P, using the word "wife" in the context of "took an additional wife". For Numbers:13:21-22 (P, misclassified as J), the model misses the discontinuity introduced between the preceding and succeeding spans, whose specific focus on "grapes" is strangely interrupted (though this feature is also inaccessible due to the initial feature selection). Finally, Deuteronomy:32:48-52 (O, misclassified as P) is interesting because it is a direct copy of Numbers:27:12-14, which is indeed P.

## 4 Future work

Along with graph convolutional networks, several graph-aware neural models have recently been introduced (e.g. graph attention networks (Veličković et al., 2017), tree-structured variational autoencoders (Yin et al., 2018)), and their effectiveness should be tested on this task. In particular, vanilla GCNs are limited in how they integrate information from other nodes, and the expressivity of these models may prove useful for the more complex relationships involved in compositional forces. Active research into augmented GCNs (Lee et al., 2018) is another avenue for addressing the current limitations.

[3]One of the common nouns that met the filter criterion.

| Span | True | Guess | Diff |
|---|---|---|---|
| Exodus:14:8 | P | R | 88.42 |
| Numbers:13:21-22 | P | J | 88.10 |
| Genesis:37:28.11-20 | J | P | 83.88 |
| Genesis:30:4.1-6 | J | P | 81.32 |
| Deuteronomy:32:48-52 | O | P | 78.96 |
| Genesis:21:2.1-6 | J | P | 66.48 |
| Genesis:25:1-4 | E | P | 62.54 |
| Numbers:26:9-11 | R | P | 60.95 |
| Exodus:14:25.1-6 | E | J | 60.61 |
| Genesis:22:11.1-16.5 | R | J | 59.36 |

Table 4: Top ten misclassifications based on difference between the probability of the true label and the probability of the (incorrectly) guessed label

There are existing resources for Hebrew NLP (Multiple, 2019) that, in principle, could facilitate feature engineering. Authors often have strong positive or negative dispositions regarding people, places, activities, and the like. Moses vs. Aaron is the most obvious for the DH, but characters like Baalam and many of the pre-exilic judges/kings have striking mixtures of praise and condemnation. Sentiment detection (Amram et al., 2018) might provide a window into these differences. Several DH justifications involve concept-realization (most famously, the use of Elohim vs Jehovah for the Deity), and being able to tie two words as alternate expressions of the same concept would be very useful. However, we are hesitant to incorporate modern resources due to potential bias, both in general language (given Hebrew's long existence as a liturgical language and subsequent revival) and specific resources created by scholars who may unintentionally encode their own conclusions. We therefore are experimenting with training unsupervised distributional models (Blei et al., 2003; Mikolov et al., 2013; Lippincott et al., 2012; Rasooli et al., 2014) directly on Biblical and contem-

porary texts to produce low-bias probablistic linguistic resources.

There is a far richer space of traditional scholarly hypotheses regarding the Bible that we plan to consider in future work. For example, the Deuteronomist sources are historically entangled with the historical books (Judges through Kings), and the prophet Jeremiah and his scribe, Baruch, which ties them to a number of spans outside the Torah (Friedman, 1987). Other annotations include: spans thought to be written in the closely-related Aramaic language, links between narrative doublets, information on poetic meter, and observations on antiquated linguistic markers. We are augmenting the initial TEI document with these annotation layers.

We framed our task as supervised span classification of a source-critical hypothesis, with the spans themselves (and hence their structural relations) taken for granted. Our longer-term goal is hypothesis *generation*, in which a model can be applied to unseen documents and propose their compositional structure. This will involve combining a linguisticly-driven model with a structural model that encourages parsimonious hypotheses. Data for training such a structural model is an open question: version control for collaborative writing is a natural modern choice, but only partially overlaps with the phenomena in the centuries-long transmission of historical text.

## 5 Conclusion

We have demonstrated that a simple graph convolutional network outperforms graph-unaware models on a task from traditional source criticism. Our error analysis revealed several characteristic shortcomings of the model and feature set, and we discussed future directions to address these.

This study is also a first step towards a more general approach to studying compositional forces in richly-structured historical texts. The basic assumptions of a tree-structured document with traditional annotations attached to nodes fits many situations, and in fact an immediate next step is to adopt these procedures to arbitrary TEI-encoded data sets and metadata. This will open up a broad range of existing documents and hypotheses (Smith et al., 2000; Tom Elliott, 2017; Association for Literary and Linguistic Computing, 1977; University of Ulster, 2017), and encourage collaboration with domain experts via e.g. common visualization and annotation tools.

## References


Adam Amram, Anat Ben-David, and Reut Tsarfaty. 2018. Representations and Architectures in Neural Sentiment Analysis for Morphologically Rich Languages: A Case Study from Modern Hebrew. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2242–2252.

Association for Literary and Linguistic Computing. 1977. Oxford Archive of Electronic Literature.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent Dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

TEI Consortium editors. 2019. TEI P5: Guidelines for Electronic Text Encoding and Interchange.

Richard Elliott Friedman. 1987. *Who Wrote the Bible?* Simon and Schuster.

Richard Elliott Friedman. 2003. *The Bible with Sources Revealed.* HarperCollins.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

John Boaz Lee, Ryan A Rossi, Xiangnan Kong, Sungchul Kim, Eunyee Koh, and Anup Rao. 2018. Higher-order graph convolutional networks. *arXiv preprint arXiv:1809.07697*.

Thomas Lippincott. 2009. A Framework for Multilayered Boundary Detection. *Digital Humanities 2009*.

Thomas Lippincott, Diarmuid O Séaghdha, and Anna Korhonen. 2012. Learning syntactic verb frames using graphical models. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volum e 1*, pages 420–429. Association for Computational Linguistics.

Kirk E. Lowery. 2016. A Reference Guide to the Westminster Leningrad Codex.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Frederick Mosteller and David L. Wallace. 1963. Inference in an Authorship Problem. *Journal of the American Statistical Association*, 58(302):275–309.

Multiple. 2019. Hebrew NLP Resources.

Martin Potthast, Francisco Rangel, Michael Tschuggnall, Efstathios Stamatatos, Paolo Rosso, and Benno Stein. 2017. Overview of PAN'17: Author Identification, Author Profiling, and Author Obfuscation. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction. 7th International Conference of the CLEF Initiative (CLEF 17)*, Berlin Heidelberg New York. Springer.

Martin Potthast, Benno Stein, Alberto Barrón-Cedeño, and Paolo Rosso. 2010. An evaluation framework for plagiarism detection. In *Proceedings of the 23rd international conference on computational linguistics: Posters*, pages 997–1005. Association for Computational Linguistics.

Mohammad Sadegh Rasooli, Thomas Lippincott, Nizar Habash, and Owen Rambow. 2014. Unsupervised Morphology-Based Vocabulary Expansion. In *ACL (1)*, pages 1349–1359.

Yunita Sari, Mark Stevenson, and Andreas Vlachos. 2018. Topic or Style? Exploring the Most Useful Features for Authorship Attribution. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 343–353.

Shachar Seidman and Moshe Koppel. 2017. Detecting pseudepigraphic texts using novel similarity measures. *Digital Scholarship in the Humanities*, 33(1):72–81.

Yanir Seroussi, Ingrid Zukerman, and Fabian Bohnert. 2011. Authorship attribution with latent Dirichlet allocation. In *Proceedings of the fifteenth conference on computational natural language learning*, pages 181–189. Association for Computational Linguistics.

DA Smith, JA Rydberg-Cox, and GR Crane. 2000. The Perseus Project: a digital library for the humanities. *Literary and Linguistic Computing*, 15(1):15–25.

Efstathios Stamatatos. 2009. A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology*, 60(3):538–556.

Kalaivani Sundararajan and Damon Woodard. 2018. What represents "style" in authorship attribution? In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2814–2822.

Hugh Cayless et al. Tom Elliott, Gabriel Bodard. 2017. EpiDoc: Epigraphic Documents in TEI XML.

University of Ulster. 2017. CELT: Corpus of Electronic Texts.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Pengcheng Yin, Chunting Zhou, Junxian He, and Graham Neubig. 2018. StructVAE: Tree-structured latent variable models for semi-supervised semantic parsing. *arXiv preprint arXiv:1806.07832*.