Semantic Vector Combinations and the Synoptic Gospels

Dominic Widdows[1] and Trevor Cohen[2]

[1] Google, Inc. (widdows@google.com)
[2] Arizona State University (trevor.cohen@asu.edu)

Abstract. This paper applies some recent methods involving semantic vectors and their combination operations to some very traditional questions, including the discovery of similarities and differences between the four Gospels, relationships between individuals, and the identification of geopolitical regions and leaders in the ancient world. In the process, we employ several methods from linear algebra and vector space models, some of which are of particular importance in quantum mechanics and quantum logic.

Our conclusions are in general positive: the vector methods do a good job of capturing well-known facts about the Bible, its authors, and relationships between people and places mentioned in the Bible. On the more specific topic of quantum as opposed to other approaches, our conclusions are more mixed: on the whole, we do not find evidence for preferring vector methods that are directly associated with quantum mechanics over vector methods developed independently of quantum mechanics. We suggest that this argues for synthesis rather than division between classical and quantum models for information processing.

1 Introduction

Semantic vector approaches have been used with considerable research success in recent years. Applications have included information retrieval, automatic word sense discrimination, ontology acquisition, and the creation of practical aids to document annotation and translation.

During the recent period in which these tools have been developed, most empirical research in computational linguistics has been devoted to large and rapidly growing corpora. This is for very good reasons. Many current information needs are greatest when dealing with the recent explosion in the scale of available information. The rapidity with which information sources such as the World Wide Web have developed has forced the adoption of new information search and exploration strategies, some not previously possible or necessary.

At the same time, much cultural and literary scholarship focusses (appropriately) on comparatively small and well organized corpora — studying (for example) works that have long been established as scriptures and classics. Resources in the form of concordances, cross references, and commentaries, have been readily available in paper form for many of the scriptures and classics for some centuries, and these information modalities are very much the prototypes for today's electronic indexes, hyperlinks, and commenting, tagging, annotation and collaborative filtering systems.

This paper tries to take a step that may be considered retrograde, or at least retrospective: to see what recent advances in empirical semantic vector analysis may have to say on some simple issues in literary scholarship, particularly Biblical scholarship. Naturally, our goal is not to discover something as yet unseen in a field which has had many careful research lifetimes already devoted to it: rather, it is to see if a very simple mathematical machine can retrieve any comparable results, and to see if this sheds any useful light on techniques of automatic information analysis more generally. In the process, we hope to demonstrate and test some recent developments in semantic vector methodology, particularly with regard to semantic combination and composition operations.

It is hoped that this latter aspect of the work presented will be of particular interest to the quantum interaction community: specifically because some of the vector combination techniques relate directly to operations used in quantum mechanics (in particular eigenvalue decomposition) and quantum logic (particularly the non-distributive disjunction). At the same time, other techniques in vector mathematics including permutation and clustering are also useful in semantic analysis. If vector operations can be largely categorized as "quantum" or "non-quantum", there seems to be no experimental reason at this stage for preferring the "quantum" over the "non-quantum" vector operations. This may help to inform the investigation of questions about the developing focus of "quantum interaction" as an area derived from quantum physics, or an area evolving at least somewhat independently, and about how this field should be characterized.

2 Methods Used

The semantic vector methods used in this paper are descendants of the vector model for information retrieval, and the subsequent development of latent semantic analysis, which compresses the sparse information in the vector model's term by document matrix into a more condensed, lower-dimensional representation. The relationship between these structures and the quantum logic of Birkhoff and von Neumann [1] has been further recognized in recent years (see particularly [2, 3]).

Particular methods used from these models include:

- Vector sum for composition, from the earliest vector model search engines [4].
- Singular Value Decomposition for more compressed semantic representation, from Latent Semantic Analysis [5].
- The use of subspaces as another more generalizing model for disjunction [2, 6, 3].
- The use of orthogonality to model complementation and negation [3, Ch. 7].

Other methods from the wider literature that are used particularly include:

- Clustering for finding more stable units (c.f., ‘quanta’) among observed results, as developed for word sense discrimination [7].
- Visualization of groups of word vectors using principal component plotting [8].
- The recent permutation based construction of semantic vectors [9].
- Pathfinder link analysis, a graph construction and visualization method [10].

The data used in our experiments is principally the King James Bible, that is, the translation into English of Jewish (Hebrew language) and Christian (Greek language) scriptures, authorised under King James (VI of Scotland, I of England), dated to 1611.

Nearly all of the software and corpora used in this paper is freely available and relatively easy to use. The corpus is from Project Gutenberg (www.gutenberg.org). Free software components are from Apache Lucene (lucene.apache.org), the Semantic Vectors project (semanticvectors.googlecode.com), and the Java Matrix Package (math.nist.gov/javanumerics/jama) used for singular value decomposition.

3 Semantic Vectors and the Synoptic Gospels

This section describes our single most deliberate experiment: testing to see if vector analysis discerns the similarity of the Synoptic Gospels. Since at least the second century AD, the Christian writings gathered in the New Testament have included four canonical accounts of the activities of Jesus of Nazareth (ca. 5BC - 30AD), and these writings, called the Gospels, have since the earliest times been attributed to authors called Matthew, Mark, Luke, and John. A basic tenet of New Testament scholarship is that Matthew, Mark and Luke are closely related, with much material drawn from one another or at least from common sources. For this reason, these three Gospels are referred to as the Synoptic (Greek, “joined eye”) Gospels.

3.1 Vector Sum Similarity

In this experiment, we set out to discover whether a semantic vector model built from the text of the King James Bible shared the view that Matthew, Mark and Luke are similar and John is the odd one out. Semantic vectors for terms (frequency > 10, stopwords removed) were produced using random projection (reduced dimension $= 200$) on the Lucene term-by-document matrix, as implemented in the SemanticVectors package [11]. Random projection is a computationally efficient variant of Latent Semantic Analysis: instead of computing exactly orthogonal latent axes using Singular Value Decomposition, latent axes are chosen randomly, based on the mathematical property that randomly chosen axes can be demonstrated to be nearly orthogonal in a suitably quantifiable sense [12].

Document vectors for each chapter were produced using the (normalized) weighted vector sum of term vectors, and combined vectors for each of the four

Gospels were computed as a normalized vector sum of the document vectors representing the chapters of each book. (This latter sum is implemented on the fly using a useful regular expression matching query builder applied to the filesystem paths: this technique can be easily used for other potentially interesting aggregate queries, such as producing query terms combining many morphological variants of the same root.) Pairwise similarities between the four resulting vectors were computed, and are shown in Table 1 and Table 2. The first table shows similarities in a model computed using the entire King James Bible, the second one shows similarities in a much smaller model computed using only the Gospel texts themselves.

Table 1. Cosine similarities between the Gospels, whole Bible model

	Matthew	Mark	Luke	John
Matthew	1	0.995	0.998	0.990
Mark		1	0.996	0.987
Luke			1	0.989
John				1

Table 2. Cosine similarities between the Gospels, Gospels only model

	Matthew	Mark	Luke	John
Matthew	1	0.990	0.994	0.969
Mark		1	0.991	0.968
Luke			1	0.969
John				1

Two things are immediately apparent. Firstly, the similarities are on the whole very high. Often nearest neighbour similarities in such models range from 0.3 to 0.7 (see the Tables later in this paper for a sample of reasonably typical values), so any cosine similarity greater than 0.9 is very high. It appears that the commonalities between the Gospels (e.g., use of frequent terms) outweigh their differences by a long way. This may be due to the “bag of words” nature of the creation of document vectors. In bag of words methods, the order of words is not taken in to account — in this case, due to the commutative property of vector addition. Thus if the Gospels share many common words with typical frequencies, they will have similar document vectors. By comparison, average similarities between the Gospels and earlier Old Testament works tend to be in the range of 0.9 to 0.95 (see Table 3). It is reasonable that these are lower similarities, though they are still high, and some statistical analysis of document creation and term reuse may help to account for this.

Table 3. Cosine similarities between the Gospels and a sample of Old Testament books

	Matthew	Mark	Luke	John
Exodus	0.945	0.932	0.946	0.921
1 Kings	0.949	0.942	0.956	0.926
Psalms	0.934	0.912	0.934	0.929
Jeremiah	0.950	0.931	0.950	0.934

Secondly, even within these very close results, John is clearly the odd one out, having lower similarities with all of the other Gospels than are found in between the three Synoptic Gospels. This is particularly apparent in the smaller model, though this appears to be partly because the smaller model shows similar comparisons but distributed across a wider range of scores. We note in passing that these experiments were repeated several times with different dimensions (ranging from 100 to 1000), with remarkably similar and often exactly the same results.

3.2 Cluster Comparison of Chapters

Another way of analysing similarities and differences between the Gospels is to cluster the individual chapter vectors (instead of summing them into combined book vectors). Clustering chapters provides a much richer qualitative analysis, at a greater computational cost. However, for a dataset the size of the Gospels (89 chapter vectors), this cost is trivial in contemporary terms. The clusters in our experiments are produced using the k-means algorithm: at each stage of the algorithm, each vector is assigned to its nearest cluster centroid, and then the centroids of the clusters are recomputed based on the new assignment. An implementation of this algorithm is included in the SemanticVectors package.

The results with 20 clusters clearly demonstrate the distinctive nature of John's Gospel. The chapters of John's Gospel tend to appear in tight clusters, a majority of whose members are from the same Gospel: on the other hand, if a cluster contains chapters from one of the Synoptic Gospels, it is far more likely to include chapters from others of these Gospels. A simple quantitative measure of the distinct nature of John's Gospel can be obtained using conditional probability: given that one chapter in a cluster is from a particular Gospel, what is the probability that another chapter in the same cluster is from the same Gospel? Typical results obtained in this experiment were:

John: 0.66 Matthew: 0.28 Luke: 0.24 Mark: 0.18.

Note that due to the random initialization of clusters, results from clustering runs are not identical each time. In each of several runs, the score for John was above 0.5, a threshold never breached by any of the other Gospels. This shows that that the chapters in John's Gospel have, on average, stronger mutual similarities than those of the other three Gospels, which are much more easily mixed together.

A further interesting experiment would be to extend the cluster analysis to cover pairwise conditional probabilities, to see if these reflect the known patterns of how the Synoptic Gospels borrowed from each other.

It is interesting to note that authorship, though important, is only one variable that influences similarity in our results. Sometimes describing similar content is more clearly responsible for similarity: for example, the four element cluster {Luke 23, Matthew 27, John 19, Mark 15} appears in several experimental runs, and each of these four chapters contains the author's account of the crucifixion.

We may conclude that, when asked "Which of the Gospels are similar?", the vector model answers "Matthew, Mark and Luke are similar, John is a bit different", but the model is also sensitive to factors other than authorship, that sometimes produce stronger affinities between texts.

4 Visualization of Disjunctions

This section describes experiments in visualizing the effects of different combination operations on search results. Lists of related terms to the query "jesus + abraham + moses" were obtained using three different query building and search ranking methods:

1. Vector sum of the constituent vectors. (Figure 1.)
2. Quantum disjunction of the constituent vectors: that is, results are ranked according to their proximity to the subspace spanned by the query vectors. (Figure 2.)
3. Minimum distance (maximum similarity) to any one of the constituent vectors. (Figure 3.)

The search results are projected down to 2 dimensions by computing the singular value decomposition (using the Jama package) and by plotting the vectors according to the second and third coordinates of their reduced vectors (the first component often mainly says "all the data is somewhere over here in the semantic space" [8]). The plotting itself is performed using a small Java Swing utility from SemanticVectors.

On analysis, the main distinction in the results is between the maximum similarity method and the other two. The maximum similarity method produces, as expected, several results that are similar to just one of the constituents, rather than their more general combination. For example, many of the close relatives and associates of Abraham make it into the minimum distance results, whereas only his wife Sarah is present in the other results.

While the other two results sets have much in common, including many more general terms, there is a small suggestion that the disjunction similarity preserves some of the close neighbours as well as the more general terms, for example, Moses' brother Aaron appears in the disjunction results and not the vector sum results. This is something we should have expected, since with the quantum disjunction, if an element is close to one of the generators of a subspace, it will naturally be close to the subspace generated.

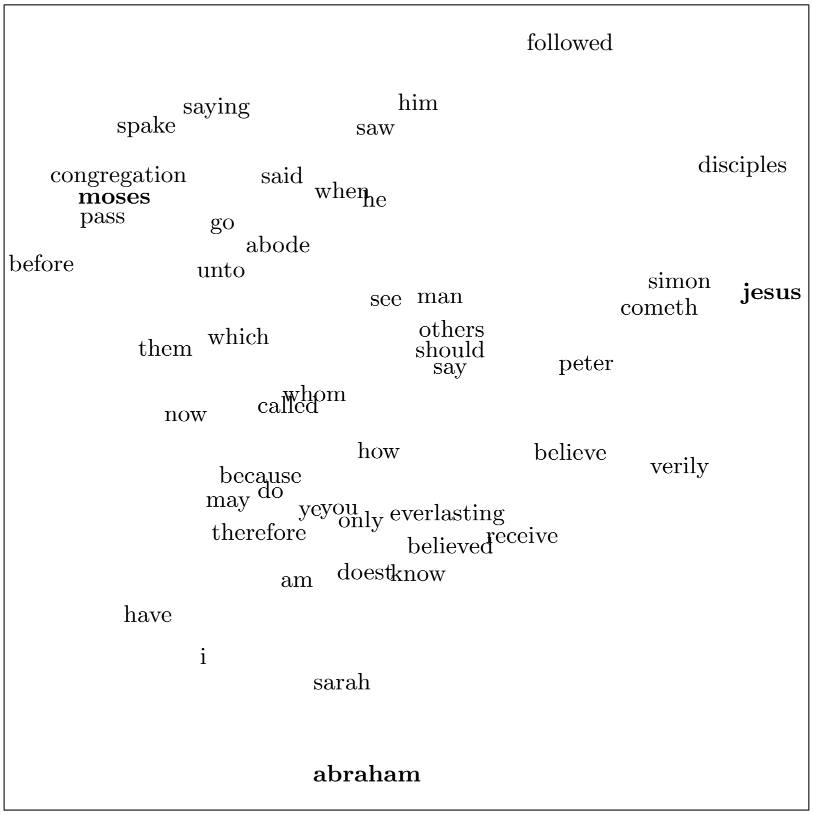

Fig. 1. Neighbours of "Jesus", "Moses" and "Abraham", using Vector Sum

5 Permutation Similarity

Our final set of experiments uses the permutation indexing method developed by Sahlgren et al [9], and demonstrates that this method is a powerful enhancement over raw vector similarity at the task of extracting the names of ancient kingdoms from the Bible corpus. In essence, the permutation method works by indexing each term not only as a sum of terms in the surrounding context, but as a *permuted* sum, the permutation in coordinates being governed by the relative positions of the words in question. (A more geometric interpretation can be obtained by noting that many permutations of coordinates are effectively rotations in the semantic space.)

Table 4 shows that, in these cases, results from the permutational query (left hand column) are much more specific in their relationships than those of traditional similarity queries (center and right hand column). In the permutation results, the query "king of ?" finds fillers for the target "?" based on cosine similarity with the permuted vectors for "king" and "of", and picks out purely

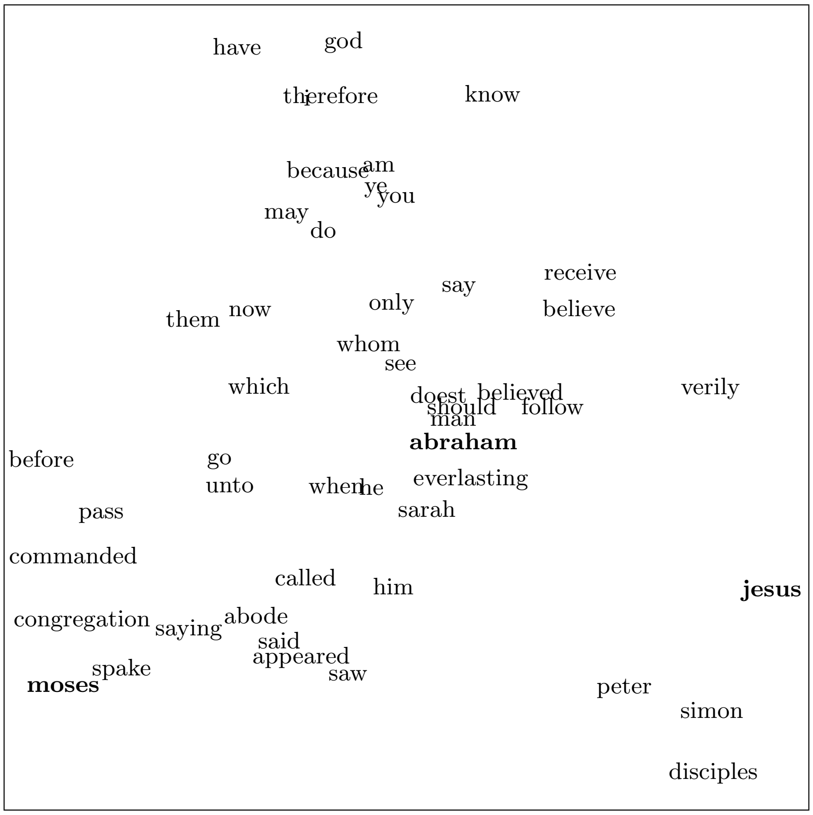

Fig. 2. Neighbours of "Jesus", "Moses" and "Abraham", using Vector Subspace

the names of geopolitical regions in the ancient world. By contrast, if we were to try and construct such a list using traditional cosine similarity, either with a seed example such as "assyria" or one of the same query terms, "king", the results are much less accurate.

Note that in the results presented here, the permutation model was built without removing stopwords (which preserves the integrity of patterns based on exact word order), whereas the similarity results were obtained by removing stopwords as usual. From studying examples, we believe this choice is optimal for each model so makes for a reasonably fair comparison.

As the "king of ?" permutation query illustrates, near neighbours in permutation derived spaces tend to be of the same semantic type (in this case, they are all kingdoms). However, these neighbours need not be thematically related. For example, a query for "adam" in a permutation-based space retrieves the cast of (male) biblical characters in the left-hand column of Table 5. Several of these characters neither appear together in the scripture, nor are they genealogically related.

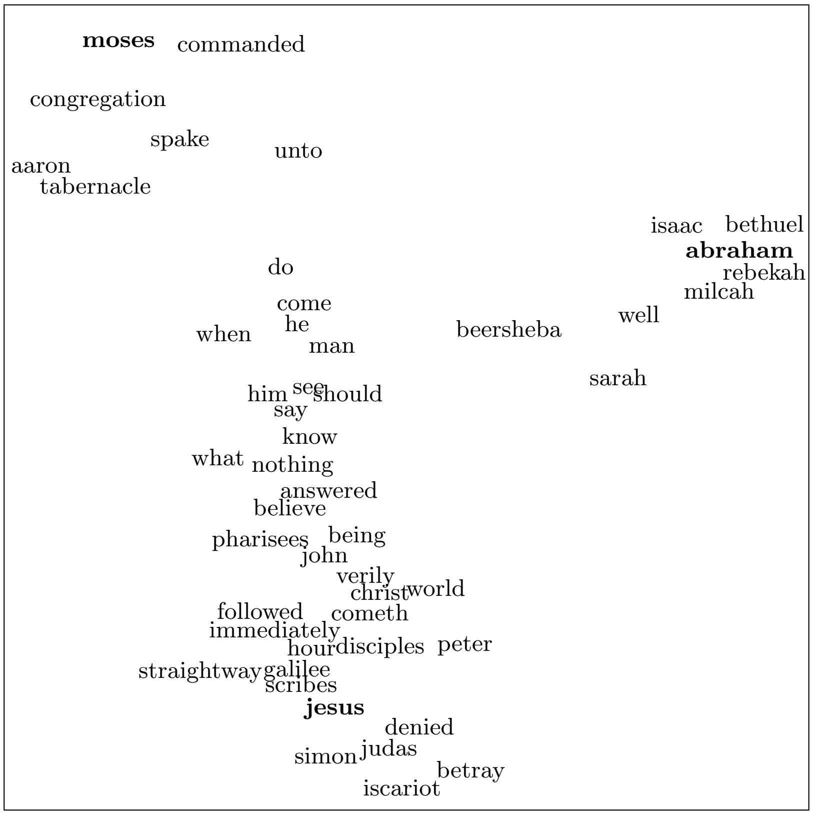

Fig. 3. Neighbours of "Jesus", "Moses" and "Abraham", using Maximum Similarity

In contrast, the nearest neighbours of "adam" in a vector space constructed using term-document statistics without regard for word order appear in the right-hand column of Table 5. While these results do include biblical characters (some of Adam's descendants), other elements of the Story of the Fall are also included.

These two types of indexing capture different types of relations between terms. Moreover, it is possible to construct a vector space that combines these relations by using trained (rather than random) term vectors as the basis for a permutation-based space. Each term is then indexed as the coordinate-based permuted sum of a set of meaningful term vectors. This hybrid vector space includes both thematic and order-based associations, supporting a simple sort of inference: queries for "queen ?" retrieve the names of kings as well as queens (see Table 6).

Another way to combine the strengths of these types of indexing procedures is to use the associations generated with one indexing procedure to evaluate relations between nearest neighbours generated in the other. This combination

Table 4. Permutation and similarity results for geopolitical entities

Permutation query “*king of ?*”	
0.728	assyria
0.699	babylon
0.662	syria
0.647	zobah
0.604	persia
0.532	judah

Similarity query “*king*”	
1.000	king
0.441	province
0.408	reign
0.380	had
0.378	did
0.377	came

Similarity query “*assyria*”	
1.00	assyria
0.653	sennacherib
0.628	rabshakeh
0.626	hezekiah
0.575	hoshea
0.509	amoz

Table 5. Neighbours of “adam” in different semantic spaces

Permutation-based space (word order encoded)	Term-document space (word order ignored)
1.00 adam	1.00 adam
0.676 joseph	0.552 enoch
0.654 saul	0.518 garden
0.641 aaron	0.505 lamech
0.639 noah	0.444 eden
0.638 david	0.407 sixty

Table 6. Combining order-based and order-agnostic vectors

Search for “king ?”		Search for “queen ?”	
Random vector basis	Term vector basis	Random vector basis	Term vector basis
0.756 ahasuerus	0.604 ahasuerus	0.187 desiring	0.332 vashti
0.643 agrippa	0.571 agrippa	0.184 exhort	0.314 ahasuerus
0.493 ahaz	0.465 rehoboam	0.181 whithersoever	0.302 agrippa
0.464 rehoboam	0.451 ahaz	0.172 vashti	0.288 absent
0.401 delighteth	0.450 delighteth	0.168 equity	0.287 darius

allows for the construction of queries such as "what thing of the same semantic type as 'abraham' is most strongly associated with him" (isaac 0.538).

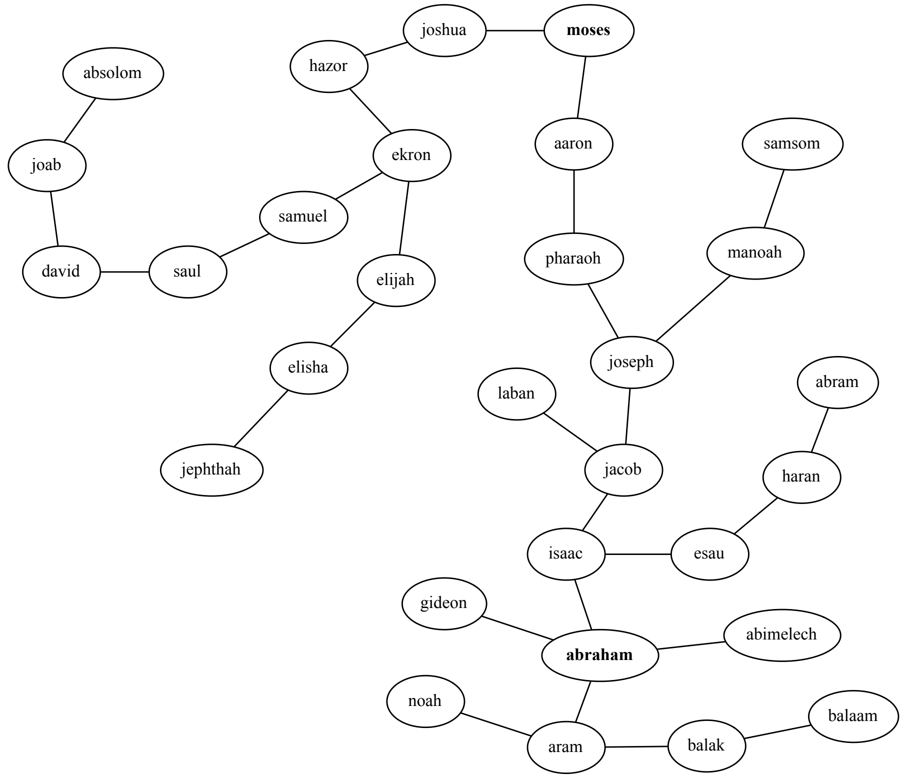

Fig. 4. Linked search results seeded with "Abraham" and "Moses"

Figure 4 illustrates the sort of information that can be extracted by combining order-based and order-agnostic representations. The nodes in the network were determined by finding the thirty nearest neighbours of the normalized sum of the vectors for the terms "abraham" and "moses" in in a permutation-based space (d=500, frequently occurring terms included). Nearest-neighbor searches in permutation-based spaces tend to produce results of the same semantic type as the search terms, in this case male biblical characters (aside from the cities Ekron and Hazor). However, these neighbours are not necessarily thematically related: many of these characters are not genealogically related, nor do they appear together in any biblical stories.

In contrast, the links in Figure 4 were determined using an order-agnostic vector space. Initially all nodes were linked according to the cosine similarity between them. The most significant links were identified using Pathfinder network scaling [10], which prunes networks such that no two nodes are linked directly if there is a shorter pathway between them via other nodes. Scaling and visualization were performed with a specially provided version of the Pathfinder software package, presently under development by Roger Schvaneveldt (the diagram has been redrawn by hand in Figure 4 for presentation in print). Pathfinder has preserved several genealogical links, such as the subtree linking Abraham, Isaac, Esau, Jacob and Joseph, and the link between Moses and Aaron. Other personal relationships are also preserved. Elijah is linked to his disciple Elisha, Saul is linked to his successor David, and Absalom is linked to his murderer, Joab. The development of further methods to combine these types of vector spaces is likely to be a fertile area for future research.

The connections from "pharaoh" to the terms "aaron" and "moses" on the one hand and "joseph" on the other are of particular interest as it indicates that the vector representation for the term "pharaoh" refers to at least two distinct individuals. Two different Pharaohs, generations apart from one another, were involved with these different characters. As is the case with ambiguous terms, it is possible to use quantum negation [3, Ch 7] to isolate different senses of a particular vector representation, as illustrated in Table 7. Initially (leftmost column), the vector representation for "pharaoh" is dominated by elements of the biblical story in which Joseph averts famine in Egypt by interpreting the dreams of the Pharaoh. Subsequently (second column from the left) the component of the vector representation of "pharaoh" that is orthogonal to the vector representation of "joseph" is isolated and normalized. In this representation, elements of the story of the Exodus from Egypt such as plagues of "flies", "frogs" and "locusts" appear in the list of nearest neighbours. As further elements of Joseph's story are removed (rightmost columns), the terms related to the Exodus improve their rankings in the list of near neighbours.

Other investigations in capturing word-order influences in semantic space models include experiments using tensor products in the SemanticVectors system [13] and convolution products using the BEAGLE system. BEAGLE uses convolution products to obtain representations to encode term position that are close-to-orthogonal to the term vectors from which they are derived. They are also reversible such that this information can be decoded. As shown by Sahlgren et al [9], both of these conditions are also met by permutation of sparse random vectors, though research comparing such approaches is still in its infancy.

The high quality of the permutation results raises the question of how they compare to results obtainable by n-gram modelling [14, Ch 6]. The ability to retrieve the names of kings as well as queens for the query "queen ?" suggest that the vector permutation method generalizes slightly compared with raw n-grams, and perhaps behaves more like a smoothed adaptation of the basic n-gram model. The comparison between n-grams and vector permutations would be fruitful to investigate further, especially since the tradeoffs between exact deduction and intelligent induction are central in discussing the relative usefulness of classical versus quantum logic (see for example [6]).

Table 7. Teasing apart ambiguous pharaoh using quantum negation

pharaoh	pharaoh NOT joseph	pharaoh NOT joseph famine
1.000 pharaoh	0.839 pharaoh	0.783 pharaoh
0.626 egypt	0.501 magicians	0.514 hardened
0.616 favoured	0.492 hardened	0.497 magicians
0.562 kine	0.488 egypt	0.419 egypt
0.543 joseph	0.442 kine	0.362 egyptians
0.543 magicians	0.432 favoured	0.358 enchantments
0.523 ill	0.358 ill	0.333 flies
0.504 dreamed	0.347 egyptians	0.326 frogs
0.499 famine	0.340 river	0.326 kine
0.452 food	0.324 land	0.317 river
0.439 dream	0.324 famine	0.307 favoured
0.435 land	0.315 plenteous	0.305 intreat
0.425 hardened	0.308 plenty	0.290 stretch
0.420 plenty	0.300 enchantments	0.252 rod
0.378 seven	0.296 flies	0.251 locusts
0.368 egyptians	0.290 stretch	0.243 plenteous
0.361 goshen	0.287 frogs	0.242 dream
0.360 plenteous	0.279 seven	0.240 hail
0.340 river	0.273 dream	0.237 houses
0.327 interpreted	0.269 locusts	0.234 ill

6 The Relevance of this work to Quantum Interaction

If our goal was to produce evidence that quantum mechanics and logic provides a correct model for natural language semantics, and classical mechanics and logic provides a flawed model, then it may be argued that these experiments are a failure. We have not (for example) demonstrated that the vectors and similarities used to model natural language violate Bell's inequalities, or that the correct combination techniques for vectors necessarily involve entanglement. While we have used the quantum disjunction and eigenvalue decompositions to good effect, there is as yet no solid ground for always preferring the quantum disjunction to one of the other options, or for viewing the eigenvalue decomposition as the single correct way to obtain distinct meanings corresponding to pure states. Thus far, these appear to be useful tools, and other tools such as clustering and permutation appear to be equally valuable, and sometimes more valuable, in analysing semantic phenomena.

However, we do not believe that this is a failure: it is not our goal to demonstrate that classical is wrong and quantum is right, any more than to demonstrate that quantum is wrong and classical is right. What we believe these experiments demonstrate is that a range of tools, drawn from the same mathematical substratum as those of quantum theory, can be usefully applied to provide relatively simple models of semantic phenomena which, in spite of their simplicity, usefully parallel the findings of human scholars. Natural language (and cognition in general) is often very complex: however, we believe our results demonstrate that some reasonable approximation to this subtlety can be obtained using mathematical tools whose history and development is closely intertwined with the methods of quantum theory. If we accept the loose generalisation that classical mechanics promotes deterministic rationalism and quantum mechanics promotes probabilistic empiricism, then our experiments demonstrate that the quantum family of approaches has much to offer, even in small and tightly encapsulated domains such as the analysis of Biblical texts.

We do not think these experiments promote quantum models as a singularly privileged path forward: rather, we think our work demonstrates that the tension between classical and quantum methods is a useful dialectic that encourages synthesis.

7 Conclusions

We have demonstrated that semantic vector methods, using the same underlying mathematical models as those of quantum theory, produce reasonable results when faced with very traditional literary tasks: in particular, analysing the relationships between the Gospel writers, and identifying geopolitical entities in the ancient world. While it is no surprise that this can be done (none of our findings are new), it is somewhat startling that it can be done based on such simple mathematical assumptions.

As well as the dialectic between classical and quantum approaches to semantic analysis, we believe our work highlights an often underappreciated potential for communication between large scale empirical approaches to analysing information (typified by new fields such as information retrieval and machine learning), and the more traditional literary approach to small scale works that are deemed to be particularly important. New developments in information retrieval and machine learning will hopefully provide tools that promote fresh analysis of important texts: meanwhile, the tradition of literary scholarship may provide deep knowledge, encouraging empirical researchers to ask more significant questions with a richer sense of what sorts of relations may be analyzed.

Acknowledgements

The first author would like to give thanks for years of teaching and influence from Kit Widdows (1946–2007), whose passion for St John's Gospel is manifested in the novel "Fourth Witness" [15].

References

1. Birkhoff, G., von Neumann, J.: The logic of quantum mechanics. Annals of Mathematics **37** (1936) 823–843
2. van Rijsbergen, C.: The Geometry of Information Retrieval. Cambridge University Press (2004)

3. Widdows, D.: Geometry and Meaning. CSLI publications, Stanford, California (2004)
4. Salton, G., McGill, M.: Introduction to modern information retrieval. McGraw-Hill, New York, NY (1983)
5. Landauer, T., Dumais, S.: A solution to Plato's problem: The latent semantic analysis theory of acquisition. Psychological Review **104**(2) (1997) 211–240
6. Widdows, D., Higgins, M.: Geometric ordering of concepts, logical disjunction, learning by induction, and spatial indexing. In: Compositional Connectionism in Cognitive Science, Washington, DC, AAAI Fall Symposium Series (October 2004)
7. Schütze, H.: Automatic word sense discrimination. Computational Linguistics **24**(1) (1998) 97–124
8. Widdows, D., Cederberg, S., Dorow, B.: Visualisation techniques for analysing meaning. In: Fifth International Conference on Text, Speech and Dialogue. Lecture Notes in Artificial Intelligence 2448, Brno, Czech Republic, Springer (September 2002) 107–115
9. Sahlgren, M., Holst, A., Kanerva, P.: Permutations as a means to encode order in word space. In: Proceedings of the 30th Annual Meeting of the Cognitive Science Society (CogSci'08), Washington D.C. (2008)
10. Schvaneveldt, R.W.: Pathfinder associative networks: studies in knowledge organization. Ablex Publishing Corp, Norwood, NJ, USA (1990)
11. Widdows, D., Ferraro, K.: Semantic vectors: A scalable open source package and online technology management application. In: Proceedings of the sixth international conference on Language Resources and Evaluation (LREC 2008), Marrakesh, Morroco (2008)
12. Papadimitriou, C.H., Tamaki, H., Raghavan, P., Vempala, S.: Latent semantic indexing: A probabilistic analysis. J. Comput. Syst. Sci. **61**(2) (2000) 217–235
13. Widdows, D.: Semantic vectors products. In: Proceedings of the Second International Symposium on Quantum Interaction, Oxford, UK (2008)
14. Manning, C.D., Schütze, H.: Foundations of Statistical Natural Language Processing. The MIT Press, Cambridge, Massachusetts (1999)
15. Widdows, K.: Fourth Witness. Writersworld Limited (2004)