Complex Coordinate-Based Meta-Analysis with Probabilistic Programming

Valentin Iovene, Gaston Zanitti, Demian Wassermann

Complex Coordinate-Based Meta-Analysis with Probabilistic Programming

Valentin Iovene, Gaston Zanitti, Demian Wassermann
INRIA Saclay, CEA, Université Paris-Saclay,
bat 145, CEA Saclay, 91191 Gif-sur-Yvette, France

Abstract

With the growing number of published functional magnetic resonance imaging (fMRI) studies, meta-analysis databases and models have become an integral part of brain mapping research. Coordinate-based meta-analysis (CBMA) databases are built by automatically extracting both coordinates of reported peak activations and term associations using natural language processing (NLP) techniques. Solving term-based queries on these databases make it possible to obtain statistical maps of the brain related to specific cognitive processes. However, with tools like Neurosynth, only single-term queries lead to statistically reliable results. When solving richer queries, too few studies from the database contribute to the statistical estimations. We design a probabilistic domain-specific language (DSL) standing on Datalog and one of its probabilistic extensions, CP-Logic, for expressing and solving rich logic-based queries. We encode a CBMA database into a probabilistic program. Using the joint distribution of its Bayesian network translation, we show that solutions of queries on this program compute the right probability distributions of voxel activations. We explain how recent lifted query processing algorithms make it possible to scale to the size of large neuroimaging data, where state of the art knowledge compilation (KC) techniques fail to solve queries fast enough for practical applications. Finally, we introduce a method for relating studies to terms probabilistically, leading to better solutions for conjunctive queries on smaller databases. We demonstrate results for two-term conjunctive queries, both on simulated meta-analysis databases and on the widely-used Neurosynth database.

1 Introduction

The non-invasivity of functional magnetic resonance imaging (fMRI) led it to dominate brain mapping research since the early 1990s (Huettel, Song, and McCarthy 2008). In the past three decades, tens of thousands of published studies acquired and analysed fMRI signals to produce hypotheses on the relationship between neural activity and cognitive processes. Quickly, the idea of meta-analysing their reported activations came to mind. Meta-analysis techniques from other fields were adapted and applied to the specific case of neuroimaging studies. Soon, an ecosystem of coordinate-based meta-analysis (CBMA) databases and tools was brought to life (e.g. Laird, Lancaster, and Fox 2005; Yarkoni et al. 2011) and became an integral part of brain mapping research. CBMA databases contain both natural language processing (NLP) features estimated from a corpus of studies, and extracted coordinates of activation peaks reported by studies. These tools are used to derive activation patterns (e.g. Wager et al. 2013; Cole et al. 2012) or reveal meaningful cognitive processes through reverse inference (e.g. Smallwood and Schooler 2015; Seghier 2013; Chang et al. 2013; Andrews-Hanna, Smallwood, and Spreng 2014). With them, researchers can define more robust regions of interest supported by past literature.

Nonetheless, currently available tools are limited in the complexity of queries that they can run against CBMA databases. Term-based queries with Neurosynth (Yarkoni et al. 2011) result in underpowered meta-analyses when queries combine multiple terms of interest. This happens because the number of studies matching queries decreases with their complexity. We believe that, with the right approach, more could be wrung out of this type of data.

Recently, NeuroQuery (Dockès et al. 2020) made it possible to produce forward inference brain maps from unstructured text-based queries. They fit a semantic-smoothing matrix to encode the relations between terms in their vocabulary. They then fit a regularised linear model that statistically maps 'smoothed' text features to brain locations. This makes it possible to produce brain maps for underrepresented terms (few studies exactly match the term). However, the definition of queries in NeuroQuery is distinct from the concept of queries on databases and their semantics, of statistical nature, is difficult to interpret. It is not possible to express logic-based queries on CBMA databases with NeuroQuery. The model is also not fitted to be integrated into a fully probabilistic model that could combine meta-analysis with other modalities, such as neuroanatomical and ontological knowledge.

Since the 1970s, the computer science community has been working on extending logic programming languages (Roussel 1975; Gallaire and Minker 1978) with probabilistic semantics to represent knowledge uncertainty inherent to real-world data (reviewed by De Raedt and Kimmig 2015). A wide variety of efficient approaches to answering questions (queries) from these programs were then proposed. Domain-specific language (DSL) are not new to the neu-

roscience community. The White Matter Query Language (WMQL) (Wassermann et al. 2013) was developed to help experts formally describe white matter tracts in a near-to-English syntax. To the best of our knowledge, applying these techniques to the formulation and resolution of rich probabilistic queries on CBMA databases has yet to be attempted.

Adopting a language-oriented programming (LOP) approach, we use probabilistic logic programming languages to formulate and solve rich queries on CBMA databases. This work fits into a broader project to design a DSL, coined NeuroLang, to make it easy to express and test cognitive science hypotheses that combine meta-analysis, neuroanatomical and ontological knowledge. **The work presented here focuses on the probabilistic semantics of NeuroLang and the resolution of CBMA queries.**

Contributions of this work are three-fold. Firstly, we investigate the feasibility and technicalities of applying probabilistic logic programming to CBMA-based brain mapping. We propose a way to encode a CBMA database as a probabilistic logic program based on CP-Logic (Vennekens, Denecker, and Bruynooghe 2009) on which rich CBMA queries can be answered. We translate this program to an equivalent Bayesian network representation in order to show that correct answers to probabilistic queries can be derived from its factorised joint probability distribution. Secondly, we explain how leveraging lifted query processing techniques (Poole 2003; Braz, Amir, and Roth 2005; Dalvi and Suciu 2012) allows us to scale to the large size of neuroimaging data at the voxel level. Thirdly, we propose a relaxed modeling of TFIDF features to better encode the relationship between terms and studies and show that fewer samples are needed to solve two-term conjunctive queries than traditional approaches, on simulated and real CBMA databases.

2 Background

2.1 Term-based queries on CBMA databases

An example of term-based query formulated in plain English is: "for each region of the brain, what is the probability that studies associated with both terms *insula* and *speech* report its activation?". The result of term-based queries are used in forward inference for obtaining a map of the brain related to the terms of the query.

A CBMA database of N studies with a fixed vocabulary of M terms can be represented as two matrices $\boldsymbol{X} \in \mathbb{R}^{N,M}$ and $\boldsymbol{Y} \in \{0,1\}^{N,K}$, where $\boldsymbol{X}_{ij}$ is a TFIDF feature measured for term j in study i and $\boldsymbol{Y}_{ik} = 1$ if voxel k is reported as activated in study i.[1] In practice, $\boldsymbol{Y}$ is a sparse matrix because only a small proportion of voxels are reported within a single study.

Forward inference brain maps are constructed from a probabilistic model where binary random variables A_k and T_j respectively model the activation of each voxel v_k and the association of studies to each term t_j. $\mathbf{P}[A_k|T_j]$ is the probability that voxel k activates in studies associated with the term j and $\mathbf{P}[A_k|T_{\text{insula}} \wedge T_{\text{speech}}]$ is the probability that voxel k activates in studies associated with both terms 'insula' and 'speech'.[2]

Neurosynth (Yarkoni et al. 2011) associates terms to studies by applying a threshold $\tau = 0.001$ to TFIDF features $\boldsymbol{X}$. Forward inference maps are obtained by estimating, for each voxel k,

$$\mathbf{P}[A_k|T_j] = \frac{\sum_{i=1}^{N} \boldsymbol{Y}_{ik} \mathbf{1}[\boldsymbol{X}_{ij} > \tau]}{\sum_{i=1}^{N} \mathbf{1}[\boldsymbol{X}_{ij} > \tau]} \tag{1}$$

Solving a query with a p-term conjunction, $\varphi = T_1 \wedge \cdots \wedge T_p$, is done by estimating, for each voxel k,

$$\mathbf{P}[A_k|\varphi] = \frac{\sum_{i=1}^{N} \boldsymbol{Y}_{ik} \mathbf{1}[\min(\boldsymbol{X}_{i1}, \ldots, \boldsymbol{X}_{ip}) > \tau]}{\sum_{i=1}^{N} \mathbf{1}[\min(\boldsymbol{X}_{i1}, \ldots, \boldsymbol{X}_{ip}) > \tau]} \tag{2}$$

As terms are added to this conjunction (and thus, complexity to the query), the term $\mathbf{1}[\min(\boldsymbol{X}_{i1}, \ldots, \boldsymbol{X}_{ip}) > \tau]$ goes to zero for an increasing number of studies. Rapidly, obtaining a meaningful brain map becomes infeasible due to statistically weak results. A different model that relaxes the hard thresholding of TFIDF features is proposed in section 4. Note that, solving a *disjunction* of two terms is done by replacing min with max, thereby requiring that only one of the TFIDF features passes the threshold. The more terms are added, the larger the number of studies that are included in the estimation. In that case, statistical power is thus not an issue.

2.2 Probabilistic logic programming

Before diving into how probabilistic logic programming can be used to encode a CBMA database, we give a brief introduction to those languages through the example of CP-Logic. We also define the syntactic restrictions of the subset of this language that we use in our DSL.

CP-Logic We use CP-Logic (Vennekens, Denecker, and Bruynooghe 2009) as an intermediate representation in the compilation of our DSL. In CP-Logic, programs contain rules (also called *CP-Events*) of the form

$$(h_1 : p_1 \vee \cdots \vee h_n : p_n) \leftarrow \varphi \tag{3}$$

where h_i are head predicates, p_i are probabilities such that $\sum_i p_i \leq 1$, and the rule's body φ is a first-order logic formula. All variables occurring in the head predicates h_i must also occur in φ. Such rules are interpreted as 'φ being true causes one of the atoms h_i to be true'. Which h_i becomes true is drawn from the probability distribution defined by probabilities p_i.

Syntactic restrictions Only a subset of CP-Logic is necessary to encode CBMA databases. We limit ourselves to programs with rules that are:

- deterministic rules $(h : 1) \leftarrow \varphi$, where φ is a conjunction of predicates and h a predicate;
- or probabilistic rules whose body is $\top$ (always true), which are probabilistic choices (if there is only one choice, they are probabilistic facts).

Moreover, recursivity is not permitted in the program.

[1] A study reports coordinates of peak activations (called *foci*) whose neighbouring voxels are considered to be reported as activated by the author(s) of the study.

[2] We use $\mathbf{P}[A_k|T_i, T_j]$ to denote $\mathbf{P}[A_k = 1|T_i = 1, T_j = 1]$.

3 Probabilistic CBMA databases

We now show how a CBMA database can be encoded as a probabilistic logic program that can be translated to a Bayesian network. Then, using the factorised joint probability distribution of the network, we show that we obtain the same solutions of queries described previously. Finally, we describe our approach to solving queries on this program. Mainly, we use lifted query processing strategies on probabilistic databases.

3.1 Encoding a CBMA database as a probabilistic logic program

The program of fig. 1 encodes a CBMA database. The uniform probabilistic choice on the SelectedStudy relation partitions the space of possible worlds such that each one corresponds to a particular study. VoxelReported and TermInStudy predicates encode matrices $\boldsymbol{Y}$ and $\boldsymbol{X}$. We write the program such that solving MARG tasks $\mathbf{P}\left[\text{Activation}(v)|\varphi\right]$, where φ conjuncts and/or disjuncts $\text{TermAssociation}(t_j)$ atoms, produces the probabilistic model of term-based CBMA queries described in section 2.1. For instance, when defining

$$\varphi = \text{TermAssociation(insula)} \wedge \text{TermAssociation(speech)}$$

$\mathbf{P}\left[\text{Activation}(v)|\varphi\right]$ is equivalent to the query $\mathbf{P}\left[A_k|T_{\text{speech}} \wedge T_{\text{insula}}\right]$ described previously. We show that in the next section.

3.2 Equivalence between the program of fig. 1 and the CBMA approach of section 2.1

To justify the design of the program in fig. 1, we translate it to an equivalent Bayesian network representation using the algorithm proposed by Meert, Struyf, and Blockeel (2008). The resulting Bayesian network is depicted in fig. 2 using plate-notation. To simplify the notation, we use A_k, T_n and T_m to denote random variables $\text{Activation}(v_k)$, $\text{TermAssociation}(t_n)$, and $\text{TermAssociation}(t_m)$. From the joint probability distribution defined by the Bayesian network, it can be derived that

$$\mathbf{P}[A_k, T_n, T_m] \tag{4}$$

$$= \sum_{i=1}^{N} \mathbf{P}[c^{\text{SS}} = i]\mathbf{P}[c_{ki}^{\text{VR}} = 1]\mathbf{P}[c_{ni}^{\text{TIS}} = 1]\mathbf{P}[c_{mi}^{\text{TIS}} = 1] \tag{5}$$

$$= \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{Y}_{ik}\mathbf{1}[\boldsymbol{X}_{in} > \tau]\mathbf{1}[\boldsymbol{X}_{im} > \tau] \tag{6}$$

and, similarly, that

$$\mathbf{P}[T_n, T_m] = \sum_{i=1}^{N} \mathbf{P}[c^{\text{SS}} = i]\mathbf{P}[c_{ni}^{\text{TIS}} = 1]\mathbf{P}[c_{mi}^{\text{TIS}} = 1] \tag{7}$$

$$= \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}[\boldsymbol{X}_{in} > \tau]\mathbf{1}[\boldsymbol{X}_{im} > \tau] \tag{8}$$

From these two joint probability distributions, we use Bayes law to derive the solution of the MARG task (as defined in De Raedt and Kimmig 2015)

$$\mathbf{P}[A_k|T_n, T_m] = \frac{\mathbf{P}[A_k, T_n, T_m]}{\mathbf{P}[T_n, T_m]} \tag{9}$$

which gives the formula of eq. (2), for $p = 2$.

The same can be shown for disjunctive queries $\mathbf{P}\left[A_k|T_n \vee T_m\right]$ by summing the results of 3 two-term conjunctive queries as follows

$$\begin{aligned}\mathbf{P}\left[A_k|T_n \vee T_m\right] = \mathbf{P}\left[A_k|T_n, T_m\right] + \\ \mathbf{P}\left[A_k|\neg T_n, T_m\right] + \mathbf{P}\left[A_k|T_n, \neg T_m\right]\end{aligned} \tag{10}$$

This confirms that the probabilistic program of fig. 1 is sound, as solving queries on the program leads to the statistical estimation described in the previous section.

3.3 Solving queries on probabilistic CBMA databases

We now explore query resolution techniques that scale to the size of large CBMA databases.

The estimation of a forward inference brain map for a two-term conjunction corresponds to the MARG task

$$\mathbf{P}\left[\text{Activation}(v)|\text{TermAssociation}(t_i), \text{TermAssociation}(t_j)\right]$$

We solve this task by defining two conjunctive queries (CQs)

$$Q_1(v) = \text{Activation}(v), \text{TermAssociation}(t_i), \text{TermAssociation}(t_j)$$
$$Q_2 = \text{TermAssociation}(t_i), \text{TermAssociation}(t_j)$$

such that $\frac{\mathbf{P}[Q_1(v)]}{\mathbf{P}[Q_2]}$ solves the MARG task. The numerator corresponds the joint probability of a voxel activation and the association to the two terms. The denominator corresponds to the joint probability of the associations to the two terms.

Knowledge compilation (KC) approaches do not scale to the size of neuroimaging data We implemented the program of fig. 1 in ProbLog2 (Dries et al. 2015). We observed that, when solving two-term conjunctive queries, grounding and compiling the program to probabilistic sentential decision diagrams (PSDDs) was intractable. This is due to the very large number of voxels, terms and studies modeled in the program. A brain is typically partitioned into a grid of about 230,000 2mm^3 voxels. On average, studies in the Neurosynth database report 3165 voxel activatons. There are 14371 studies and 3228 terms in the database. We tried compiling our program manually to PSDDs (Kisa et al. 2014). Despite our efforts, the resolution of queries was still too slow to be practical for real-world applications.

Lifted query processing of unions of conjunctive queries (UCQs) on probabilistic CBMA databases We leverage theoretical results which have identified classes of queries that lifted inference can solve in polynomial time. One major result is the dichotomy theorem (proven in Dalvi and Suciu 2012) which provides a procedure for checking that UCQs are liftable. This theorem is convenient because it guarantees that any query such that the lifted processing rules can apply is guaranteed to be solvable in PTIME. If the query is not liftable, we resort to KC-based resolution techniques.

$$
\begin{aligned}
(\text{TermInStudy}(t_j, s_i) : 1) &\leftarrow \top. \qquad \forall i \in N, \forall j \in M, \boldsymbol{X}_{ij} > \tau \\
(\text{VoxelReported}(v_k, s_i) : 1) &\leftarrow \top. \qquad \forall i \in N, \forall k \in K, \boldsymbol{Y}_{ik} = 1 \\
\left(\bigvee_{i=1}^{N} \text{SelectedStudy}(s_i) : \frac{1}{N} \right) &\leftarrow \top. \\
(\text{TermAssociation}(t) : 1) &\leftarrow \exists s \left(\text{TermInStudy}(t, s) \wedge \text{SelectedStudy}(s) \right). \\
(\text{Activation}(v) : 1) &\leftarrow \exists s \left(\text{VoxelReported}(v, s), \text{SelectedStudy}(s) \right).
\end{aligned}
$$

Figure 1: CP-Logic program encoding a probabilistic CBMA database. TermInStudy(t, s) models the presence of term t in study s. VoxelReported(v, s) encodes whether voxel v was reported in study s. The large SelectedStudy uniform probabilistic choice over studies makes each possible world correspond to a specific study. Activation(v) and TermAssociation(t) respectively model the activation of voxel v and the association with term t. The SUCC query $\mathbf{P}\left[\text{Activation}(v)\right]$ gives the marginal probability of activation of voxels over all studies. The MARG task $\mathbf{P}\left[\text{Activation}(t)|\text{TermAssociation(insula)}\right]$ results in a forward inference map for the term *insula*.

.

Because the language does not have probabilistic clauses and prevents recursivity, we can use its deterministic rules to construct UCQs associated with a given SUCC task $\mathbf{P}[\psi(\boldsymbol{x})]$, where $\psi(\boldsymbol{x})$ is a conjunction of intensional, extensional or probabilistic atoms.

This lifted approach makes it possible to solve conjunctive queries in a few seconds. *Extensional query plans* (see 4.1 of Van den Broeck and Suciu 2017) are obtained and evaluated to solve queries using a custom Python relational algebra engine.

4 Relating terms and studies probabilistically

The hard thresholding $\mathbf{1}[x > \tau]$ of TFIDF features x presented in section 2.1 misses studies that could be relevant to the resolution of queries. Because we are interested in solving more complex queries, in this section we explore a relaxation by introducing the soft-thresholding function

$$
\omega(x; \alpha, \tau) \coloneqq \sigma\left(\alpha(x - \tau)\right) \quad \in [0, 1] \tag{11}
$$

where σ is the logistic function and τ a threshold. As α increases, $\omega(x; \alpha, \tau)$ converges towards the hard-thresholding function $\mathbf{1}[x > \tau]$. With an appropriate α, a larger proportion of studies is included in the calculation of $\mathbf{P}\left[A_k|\varphi\right]$, giving better estimates on small databases. For example, results of two-term conjunctive $\mathbf{P}\left[A_k|T_1 \wedge T_2\right]$ and disjunctive $\mathbf{P}\left[A_k|T_1 \vee T_2\right]$ queries are estimated with

$$
\mathbf{P}\left[A_k|T_1 \wedge T_2\right] = \frac{\sum_{i=1}^{N} \boldsymbol{Y}_{ik} \omega\left(\boldsymbol{X}_{i1}; \alpha, \tau\right) \omega\left(\boldsymbol{X}_{i2}; \alpha, \tau\right)}{\sum_{i=1}^{N} \omega\left(\boldsymbol{X}_{i1}; \alpha, \tau\right) \omega\left(\boldsymbol{X}_{i2}; \alpha, \tau\right)} \tag{12}
$$

$$
\mathbf{P}\left[A_k|T_1 \vee T_2\right] = \frac{\sum_{i=1}^{N} \boldsymbol{Y}_{ik} \left(1 - \prod_{j=1}^{2} \left(1 - \omega\left(\boldsymbol{X}_{ij}; \alpha, \tau\right)\right)\right)}{\sum_{i=1}^{N} \left(1 - \prod_{j=1}^{2} \left(1 - \omega\left(\boldsymbol{X}_{ij}; \alpha, \tau\right)\right)\right)} \tag{13}
$$

More generally, $\mathbf{P}\left[A_k|\varphi\right]$ can be estimated for first-order logic formulas φ that blend conjunctions and disjunctions of boolean random variables $T_j, j \in 1, \ldots, M$. For example, if $\varphi = (T_1 \vee T_2) \wedge (T_3 \vee T_4)$, we have

$$
\mathbf{P}\left[A_k|\varphi\right] = \frac{\sum_{i=1}^{N} \boldsymbol{Y}_{ik} f(\boldsymbol{X}_{i1}, \boldsymbol{X}_{i2}) f(\boldsymbol{X}_{i3}, \boldsymbol{X}_{i4})}{\sum_{i=1}^{N} f(\boldsymbol{X}_{i1}, \boldsymbol{X}_{i2}) f(\boldsymbol{X}_{i3}, \boldsymbol{X}_{i4})} \tag{14}
$$

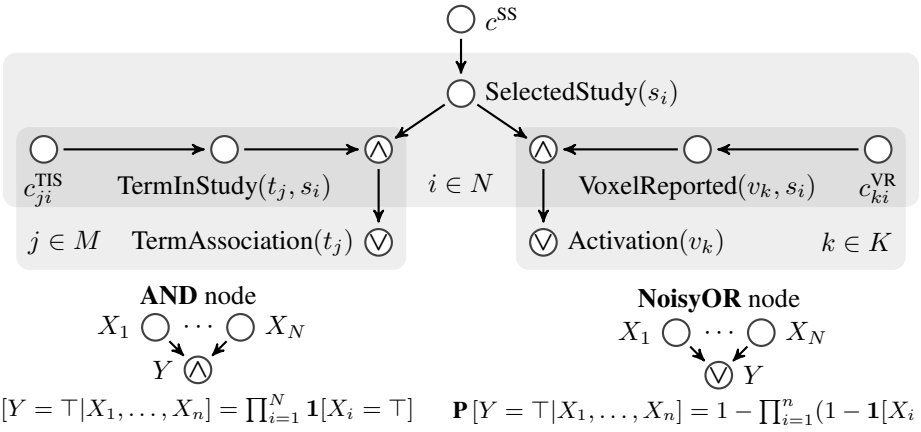
where $f(x_1, x_2) = 1 - (1 - \omega(x_1; \alpha, \tau))(1 - \omega(x_2; \alpha, \tau))$.

This modeling is implemented simply by integrating $\omega(\boldsymbol{X}_{ij}; \alpha, \tau)$ as the probabilities of probabilistic facts TermInStudy(t_j, s_i) in the program of fig. 1.

5 Experiments and results

We compare our method with Neurosynth's on simulated CBMA databases sampled from a generative model and on the Neurosynth database. Using both models, we solve 55 different two-term conjunctive queries $\mathbf{P}\left[A_k|T_i \wedge T_j\right]$.

Gain of statistical power when solving two-term conjunctive queries $\mathbf{P}\left[A_k|T_i \wedge T_j\right]$ on smaller simulated CBMA databases We evaluate our method on simulated CBMA databases obtained by sampling from the generative model



Random variables	**Domain**	**Conditional probability distribution**
c^{SS}	$\{0, \ldots, N\}$	$\forall i \in \{1, \ldots N\}, \mathbf{P}\left[c^{\text{SS}} = i\right] = \frac{1}{N}$
SelectedStudy(s_i)	$\{\top, \bot\}$	$\mathbf{P}\left[\text{SelectedStudy}(s_i) = \top \middle| c^{\text{SS}}\right] = \mathbf{1}[c^{\text{SS}} = i]$
c^{TIS}_{ji}	$\{0, 1\}$	$\mathbf{P}\left[c^{\text{TIS}}_{ji} = 1\right] = \omega(\boldsymbol{X}_{ji}; \alpha, \tau)$
TermInStudy(s_i)	$\{\top, \bot\}$	$\mathbf{P}\left[\text{TermInStudy}(s_i) = \top \middle| c^{\text{TIS}}\right] = \mathbf{1}[c^{\text{TIS}} = 1]$
c^{VR}_{ki}	$\{0, 1\}$	$\mathbf{P}\left[c^{\text{VR}}_{ki} = 1\right] = \boldsymbol{Y}_{ki}$
VoxelReported(s_i)	$\{\top, \bot\}$	$\mathbf{P}\left[\text{VoxelReported}(s_i) = \top \middle| c^{\text{VR}}\right] = \mathbf{1}[c^{\text{VR}} = 1]$

Figure 2: Plate-notation representation of the Bayesian network translated from the program described in section 3. Each ground atom in the program (e.g. TermInStudy(t_2, s_{21})) becomes a binary random variable with a deterministic conditional probability distribution (CPD). Specific **AND** nodes encode the conjunctions in the antecedent of the rules of the program. Choice random variables c^{SS}, c^{TIS}_{ji}, c^{VR}_{ki} represent probabilistic choices in the program.

of fig. 3. It provides the ground truth of which voxels activate in studies matching a given query of interest. This binary classification setting makes it possible to compare models by measuring their ability to identify these activations for multiple sample sizes. We experimented multiple numbers of voxels ($K \in [100, 10000]$). Preliminary results showed that varying the number of voxels did not alter the results. We report results for $K = 1000$ voxels, of which 5% are activated in studies matching the query. Predicted voxel activations are obtained by thresholding p-values computed from each model's estimation of $\mathbf{P}[A_k | \varphi]$ using a G-test of independence. We use a p-value threshold of 0.01 and a Bonferroni correction for multiple comparisons. Simulation results for two-term conjunctive queries are presented in fig. 4, where we compare our model's and Neurosynth's F_1 scores across 55 conjunctive queries. The F_1 score measures the performance of a binary classifier by combining its precision and recall into a single metric. We see the advantage of our approach over Neurosynth's for smaller generated samples where activations related to the query can be identified more reliably. Multiple values of α in the range $[1000, 10000]$ were tried during our experiments. We observed that the model is robust to variations of α. However when α is too small or too large, the model tends to include either too many (and irrelevant) or too few studies in the es-

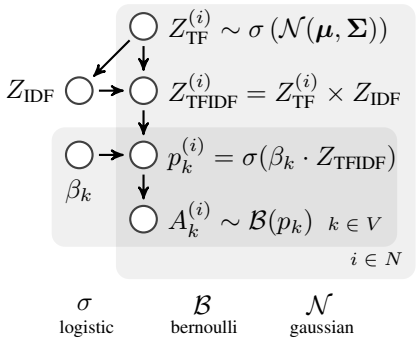


Figure 3: Model for generating CBMA databases of size N. $Z^{(i)}_{\text{TF}}$ models term frequencies in study i and follows a logistic-normal distribution. Z_{IDF} computes inverse document frequencies from $\{Z^{(i)}_{\text{TF}}\}_{i \in N}$. p_k is the probability of activation of voxel v_k. Vectors β_k are obtained from a rejection sampling scheme that controls the proportion of voxels that activate when the query is verified. Z_{IDF}, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are estimated from 4168 scrapped PubMed abstracts.

timation. We found that a sweet spot for α was around 3000 and report results for that value. Varying α around 3000 does not alter the results.

As expected, the proposed approach did not show an advantage over Neurosynth for solving two-term disjunctive queries because they do not reduce the number of studies incorporated in the estimation of $\mathbf{P}\left[A_k | T_i \vee T_j\right]$, as detailed in section 2.1.

Gain of activation consistency on a real CBMA database We evaluate our method on the Neurosynth CBMA database. Because we don't have a ground truth of which voxels activate for a given query, we resort to comparing models based on the *consistency* of their predicted activations over many random sub-samples of the Neurosynth database.

From predicted activation maps of K voxels obtained from M sub-samples of a CBMA database, the *consistency* for a two-term conjunctive formula φ is computed as

$$C_\varphi \coloneqq \frac{1}{K}\sum_{k=1}^{K}\left(1 - 2 \times \left|\frac{1}{M}\sum_{m=1}^{M}\hat{y}_{mk}^{\varphi} - \frac{1}{2}\right|\right) \qquad (15)$$

where $\hat{y}_{mk}^{\varphi} = 1$ if voxel k is predicted to be activated in sub-sample m when formula φ is true. The closer to one, the closer the average activation is to 0 or 1, which indicates a higher consistency across sub-samples. The closer to zero, the closer the average activation is to 0.5 which indicates that the predicted activations are highly variable across samples.

Results are reported in fig. 5, where the distribution of consistencies across the same 55 conjunctive queries as in section 5 are shown for multiple sample sizes. For the largest sample sizes, consistency scores are closer to 1 with our method than with Neurosynth's. For a sample size of 2395 (chosen on a logarithmic scale), the average consistency of our method was 0.48 while Neurosynth's was 0.4 (+20%) across samples and queries. For a sample size of 3856, we notice a 10% improvement.

We did not show results for larger sample sizes due to the computational cost of running the experiment on multiple large sampled databasaes, for all the conjunctive queries. Also, we were mainly interested in whether our approach would be more consistent for smaller sample sizes. We observed that the consistency between Neurosynth and our approach was similar when both models were estimated on the entire database. This means that the proposed approach is more consistent on smaller sample sizes but equivalently consistent on larger sample sizes.

6 Conclusions

6.1 Contributions

This work is a step towards incorporating rich meta-analyses in brain mapping models. We encode a CBMA database as a probabilistic logic program on which general logic-based queries can be solved. Leveraging results from lifted query processing, we are able to scale the resolution algorithms to the size of neuroimaging data. We experimented a new method for solving two-term conjunctive queries using TFIDF features more efficiently than the hard thresholding scheme used by Neurosynth. This is promising but further investigation should be conducted to know whether this method extends to queries that conjunct more terms or queries that blend conjunctions and disjunctions. The proposed method requires the same computational power than Neurosynth.

6.2 Discussion and limitations

This work fits into a broader approach to design a domain-specific language for expressing rich cognitive science hypotheses that combine neuroimaging data, neuroanatomical probabilistic maps, ontologies and meta-analysis databases to produce fine-grained brain maps supported by past literature and heterogeneous data. The flexible syntax of the language makes it possible to express all kinds of queries. $\mathbf{P}\left[\text{TermAssociation}(t) | \varphi_{\text{insula}} \vee \varphi_{\text{fMRI}}\right]$ is a reverse inference query, where φ_{insula} is a conjunction of logic predicates $\text{Activation}(v_k)$ whose probabilities come from a neuroanatomical probabilistic map of the insula, and where φ_{fMRI} is also a conjunction of predicates $\text{Activation}(v_k)$ whose voxels v_k are based on neuroimaging data coming from a custom fMRI study. Reverse inferences attempts to find the most related terms associated with a given pattern of brain activations.

The current version of NeuroLang is limited in what it can model, mainly due to the syntactic limitations on programs and queries that we had to make in order to use lifted processing strategies that scale to the size of CBMA data, as was discussed in section 3.

Recursion would make it possible to integrate spatial priors to the model. For example, to give nearby voxels the incentive to coactivate, one could define the probabilistic rule

$$\text{Activation}(v_1) : d(v_1, v_2) \leftarrow \text{Activation}(v_2) \qquad (16)$$

where d is the Euclidean distance measure between regions of the brain, normalised to be in $[0, 1]$.

The resolution of such queries remains a challenge both in terms of methodology and tractability. Future progress in the field of probabilistic programming languages could open the door to other queries of interest to the cognitive science community.

References

Andrews-Hanna, J. R.; Smallwood, J.; and Spreng, R. N. 2014. The default network and self-generated thought: component processes, dynamic control, and clinical relevance: The brain's default network. *Annals of the New York Academy of Sciences* 1316(1): 29–52. ISSN 00778923. doi: 10.1111/nyas.12360.

Braz, R. d. S.; Amir, E.; and Roth, D. 2005. Lifted First-Order Probabilistic Inference. In Kaelbling, L. P.; and Saffiotti, A., eds., *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30 - August 5, 2005*, 1319–1325. Professional Book Center. URL http://ijcai.org/Proceedings/05/Papers/1548.pdf.

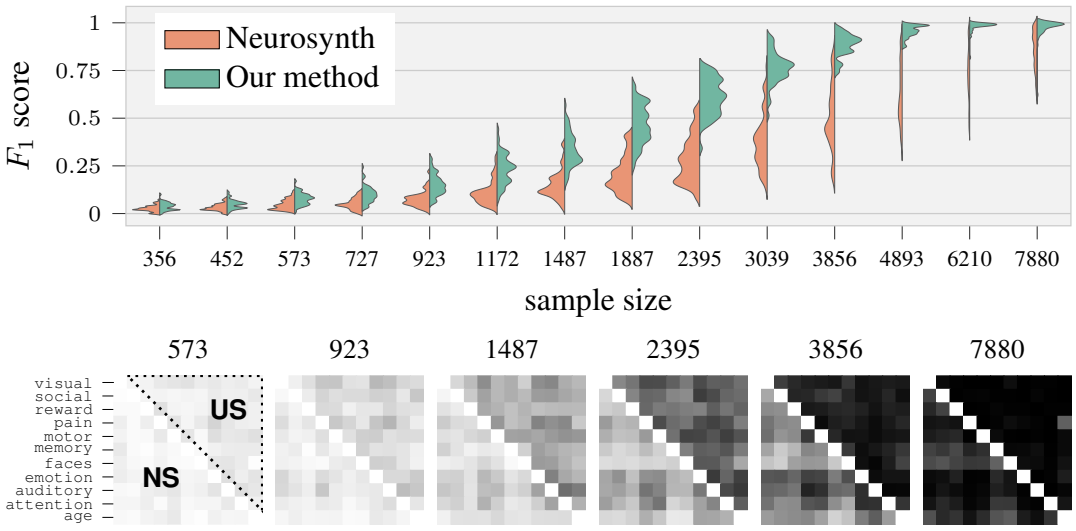


Figure 4: Comparison of Neurosynth's and our method's F_1 scores across 55 conjunctive queries $\mathbf{P}\left[A_k | T_i \wedge T_j\right]$ on simulated CBMA databases of varying sample sizes. For each sample size, 100 random sub-samples were used. **Above**, F_1 score distributions on all queries are compared across sample sizes. **Below**, F_1 score matrices (white is 0, black is 1) are compared across sample sizes. The upper triangular contains scores of our method and the lower triangular contains scores of Neurosynth. Neurosynth's default threshold $\tau = 0.001$ is used in both models. The value $\alpha = 3000$ was empirically chosen. Varying α near this value does not change the results noticeably. Sample sizes were taken on a logarithmic scale.

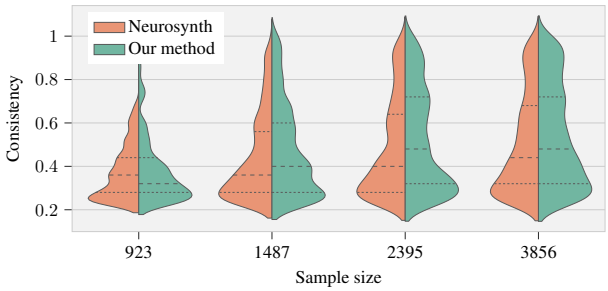


Figure 5: Comparison of both models' distributions of voxel activation consistency across 1000 sub-samples of the Neurosynth's database, for 55 two-term conjunctive queries and for multiple sample sizes. As the sample size increases, our method finds more consistent activations than Neurosynth.

Chang, L. J.; Yarkoni, T.; Khaw, M. W.; and Sanfey, A. G. 2013. Decoding the Role of the Insula in Human Cognition: Functional Parcellation and Large-Scale Reverse Inference. *Cerebral Cortex* 23(3): 739–749. ISSN 1047-3211, 1460-2199. doi:10.1093/cercor/bhs065.

Cole, M. W.; Yarkoni, T.; Repovs, G.; Anticevic, A.; and Braver, T. S. 2012. Global Connectivity of Prefrontal Cortex Predicts Cognitive Control and Intelligence. *Journal of Neuroscience* 32(26): 8988–8999. ISSN 0270-6474, 1529-2401. doi:10.1523/JNEUROSCI.0536-12.2012.

Dalvi, N.; and Suciu, D. 2012. The dichotomy of probabilistic inference for unions of conjunctive queries. *Journal of the ACM* 59(6): 1–87. ISSN 0004-5411, 1557-735X. doi:10.1145/2395116.2395119. URL https://dl.acm.org/doi/10.1145/2395116.2395119. Number: 6.

De Raedt, L.; and Kimmig, A. 2015. Probabilistic (logic) programming concepts. *Machine Learning* 100(1): 5–47. ISSN 0885-6125, 1573-0565. doi:10.1007/s10994-015-5494-z. URL http://link.springer.com/10.1007/s10994-015-5494-z.

Dockès, J.; Poldrack, R. A.; Primet, R.; Gözükan, H.; Yarkoni, T.; Suchanek, F.; Thirion, B.; and Varoquaux, G. 2020. NeuroQuery, comprehensive meta-analysis of human brain mapping. *eLife* 9: e53385. ISSN 2050-084X. doi:10.7554/eLife.53385. URL https://elifesciences.org/articles/53385.

Dries, A.; Kimmig, A.; Meert, W.; Renkens, J.; Van den Broeck, G.; Vlasselaer, J.; and De Raedt, L. 2015. ProbLog2: Probabilistic Logic Programming. In Bifet, A.; May, M.; Zadrozny, B.; Gavalda, R.; Pedreschi, D.; Bonchi, F.; Cardoso, J.; and Spiliopoulou, M., eds., *Machine Learning and Knowledge Discovery in Databases*, volume 9286, 312–315. Cham: Springer International Publishing. ISBN 978-3-319-23460-1 978-3-319-23461-8. doi:10.1007/978-3-319-23461-8_37. URL http://link.springer.com/10.1007/978-3-319-23461-8_37. Series Title: Lecture Notes in Computer Science.

Gallaire, H.; and Minker, J., eds. 1978. *Logic and data bases*. New York: Plenum Press. ISBN 978-0-306-40060-5. Meeting Name: Symposium on Logic and Data Bases, Centre d'études et de recherches de Toulouse.

Huettel, S. A.; Song, A. W.; and McCarthy, G. 2008. *Functional magnetic resonance imaging*. Sunderland, Mass: Sinauer Associates, 2nd ed edition. ISBN 978-0-87893-286-3.

Kisa, D.; den Broeck, G. V.; Choi, A.; and Darwiche, A. 2014. Probabilistic Sentential Decision Diagrams. URL https://www.aaai.org/ocs/index.php/KR/KR14/paper/view/8005/7969.

Laird, A. R.; Lancaster, J. L.; and Fox, P. T. 2005. BrainMap: The Social Evolution of a Human Brain Mapping Database. *Neuroinformatics* 3(1): 065–078. ISSN 1539-2791. doi:10.1385/NI:3:1:065. URL http://link.springer.com/10.1385/NI:3:1:065.

Meert, W.; Struyf, J.; and Blockeel, H. 2008. Learning Ground CP-Logic Theories by Leveraging Bayesian Network Learning Techniques. *Fundam. Inform.* 89: 131–160.

Poole, D. 2003. First-order probabilistic inference. In *Proceedings of the 18th international joint conference on Artificial intelligence*, 985–991.

Roussel, P. 1975. *PROLOG: Manuel de Reference et d'Utilisation*. Université d'Aix-Marseille II.

Seghier, M. L. 2013. The Angular Gyrus: Multiple Functions and Multiple Subdivisions. *The Neuroscientist* 19(1): 43–61. ISSN 1073-8584, 1089-4098. doi:10.1177/1073858412440596.

Smallwood, J.; and Schooler, J. W. 2015. The Science of Mind Wandering: Empirically Navigating the Stream of Consciousness. *Annual Review of Psychology* 66(1): 487–518. ISSN 0066-4308, 1545-2085. doi:10.1146/annurev-psych-010814-015331.

Van den Broeck, G.; and Suciu, D. 2017. Query Processing on Probabilistic Data: A Survey. *Foundations and Trends® in Databases* 7(3-4): 197–341. ISSN 1931-7883, 1931-7891. doi:10.1561/1900000052. URL http://www.nowpublishers.com/article/Details/DBS-052. Number: 3-4.

Vennekens, J.; Denecker, M.; and Bruynooghe, M. 2009. CP-logic: A language of causal probabilistic events and its relation to logic programming. *Theory and Practice of Logic Programming* 9(3): 245–308. ISSN 1471-0684, 1475-3081. doi:10.1017/S1471068409003767. URL https://www.cambridge.org/core/product/identifier/S1471068409003767/type/journal_article.

Wager, T. D.; Atlas, L. Y.; Lindquist, M. A.; Roy, M.; Woo, C.-W.; and Kross, E. 2013. An fMRI-Based Neurologic Signature of Physical Pain. *New England Journal of Medicine* 368(15): 1388–1397. ISSN 0028-4793, 1533-4406. doi:10.1056/NEJMoa1204471.

Wassermann, D.; Makris, N.; Rathi, Y.; Shenton, M.; Kikinis, R.; Kubicki, M.; and Westin, C.-F. 2013. On Describing Human White Matter Anatomy: The White Matter Query Language. In Hutchison, D.; Kanade, T.; Kittler, J.; Kleinberg, J. M.; Mattern, F.; Mitchell, J. C.; Naor, M.; Nierstrasz, O.; Pandu Rangan, C.; Steffen, B.; Sudan, M.; Terzopoulos, D.; Tygar, D.; Vardi, M. Y.; Weikum, G.; Salinesi, C.; Norrie, M. C.; and Pastor, O., eds., *Advanced Information Systems Engineering*, volume 7908, 647–654. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-642-38708-1 978-3-642-38709-8. doi:10.1007/978-3-642-40811-3_81. URL http://link.springer.com/10.1007/978-3-642-40811-3_81. Series Title: Lecture Notes in Computer Science.

Yarkoni, T.; Poldrack, R. A.; Nichols, T. E.; Van Essen, D. C.; and Wager, T. D. 2011. Large-scale automated synthesis of human functional neuroimaging data. *Nature Methods* 8(8): 665–670. ISSN 1548-7091, 1548-7105. doi:10.1038/nmeth.1635.