Compiling Tensor Expressions into Einsum

Julien Klaus, Mark Blacher, Joachim Giesen

Friedrich Schiller University Jena

Think about it!

How would you write the following expression in Python?

$$\min_{Z,B,C,D} \sqrt{\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\left(A_{ijk} - \sum_{l=1}^{L}\sum_{m=1}^{M}\sum_{n=1}^{N} Z_{lmn}B_{li}C_{mj}D_{nk}\right)^2}$$

Compiling this Expression

✓ Parse Expression

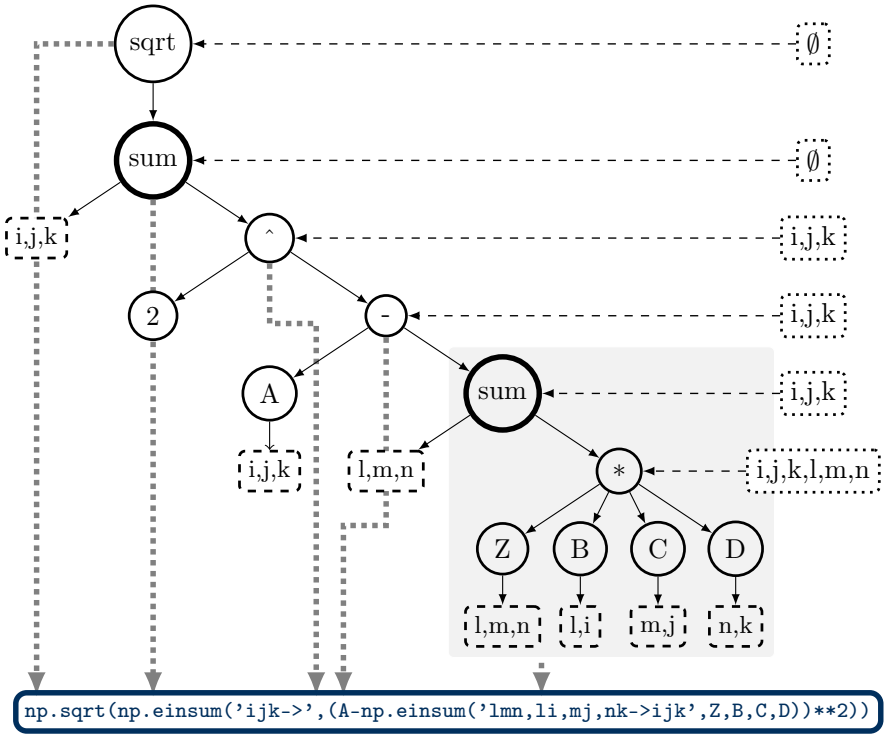
```
sqrt(sum[i,j,k]((A[i,j,k]
 -sum[n,m,l](Z[n,m,l]*B[n,i]
     *C[m,j]*D[k,l]))^2))
```

✓ Transform Tree Representation

combine multiplications and summations

✓ Generate Code

for different Python backends



Try it out

tec.pythonanywhere.com

github.com/julien-klaus/tec

Understanding Einsum Notation

Operation	Explicit expression	Einsum notation
Scalar times vector	`s*a[j]`	`,j->j`
Vector times vector	`a[i]*b[i]`	`i,i->i`
Vector outer product	`a[i]*b[j]`	`i,j->ij`
Matrix times vector	`A[i,j]*b[j]`	`ij,j->i`
Inner product	`sum[i](a[i]*b[i])`	`i,i->`
Batch matrix multiplication	`sum[k](A[b,i,k]*B[b,k,j])`	`bik,bkj->bij`
Marginalization (sum over axes)	`sum[i,l,n,o](A[i,l,m,n,o])`	`ilmno->m`
Mahalanobis distance	`sum[i,j](a[i]*A[i,j]*b[j])`	`i,ij,j->`

Acknowledgments

This work was supported by the Carl Zeiss Foundation within the project Interactive Inference (*inference.uni-jena.de*) and from the Ministry for Economics, Sciences and Digital Society of Thuringia (TMWWDG), under the framework of the Landesprogramm ProDigital (DigLeben-5575/10-9).