In [1]:

```
from __future__ import division
from __future__ import print_function

import time

import math
import torch

import numpy as np
import scipy.sparse as sp
import torch.optim as optim

import torch.nn as nn
import torch.nn.functional as F
from torch.nn.parameter import Parameter
from torch.nn.modules.module import Module
```

In [2]:

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

In [3]:

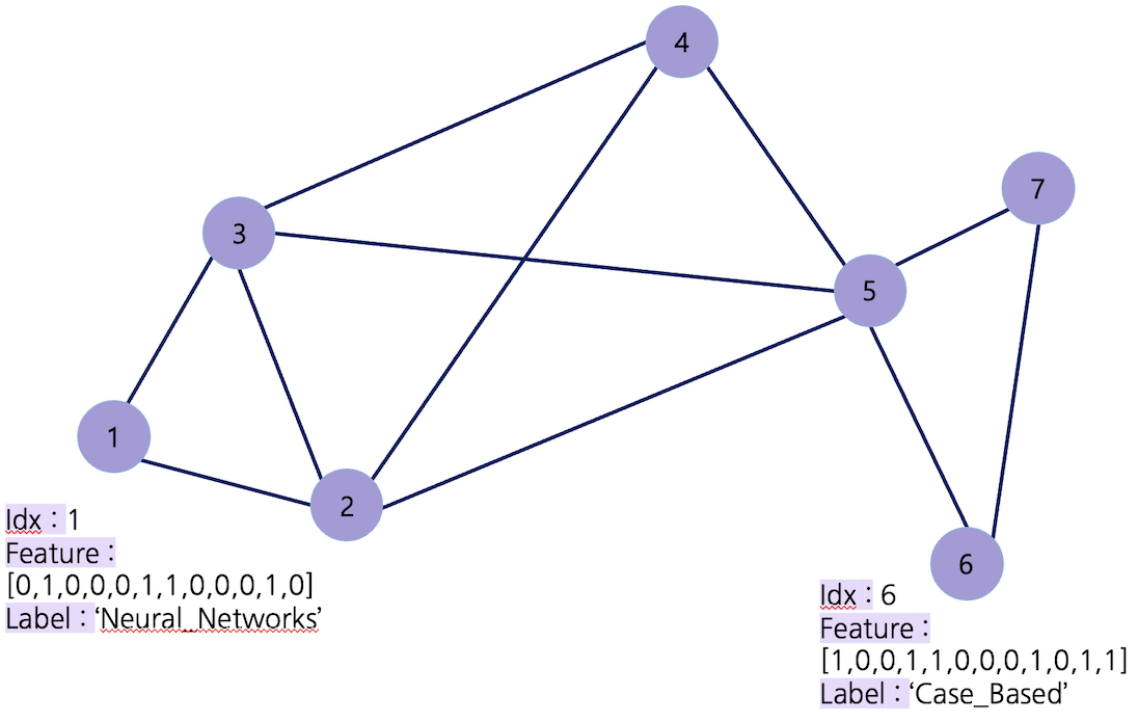
```
def encode_onehot(labels):
    classes = set(labels)
    classes_dict = {c: np.identity(len(classes))[i, :] for i, c in
                    enumerate(classes)}
    labels_onehot = np.array(list(map(classes_dict.get, labels)),
                             dtype=np.int32)
    return labels_onehot

def sparse_mx_to_torch_sparse_tensor(sparse_mx):
    """Convert a scipy sparse matrix to a torch sparse tensor."""
    sparse_mx = sparse_mx.tocoo().astype(np.float32)
    indices = torch.from_numpy(
        np.vstack((sparse_mx.row, sparse_mx.col)).astype(np.int64))
    values = torch.from_numpy(sparse_mx.data)
    shape = torch.Size(sparse_mx.shape)
    return torch.sparse.FloatTensor(indices, values, shape)

def normalize(mx):
    """Row-normalize sparse matrix"""
    rowsum = np.array(mx.sum(1))
    r_inv = np.power(rowsum, -1).flatten()
    r_inv[np.isinf(r_inv)] = 0.
    r_mat_inv = sp.diags(r_inv)
    mx = r_mat_inv.dot(mx)
    return mx
```

$$\begin{bmatrix}1&1&0\\1&1&1\\0&1&0\end{bmatrix} \begin{matrix}=2\\=3\\=1\end{matrix} \Rightarrow \begin{bmatrix}1/2&0&0\\0&1/3&0\\0&0&1/1\end{bmatrix} \times \begin{bmatrix}1&1&0\\1&1&1\\0&1&0\end{bmatrix} = \begin{bmatrix}1/2&1/2&0\\1/3&1/3&1/3\\0&1/1&0\end{bmatrix}$$

mx rowsum r_mat_inv mx 최종값

1. Graph data preprocessing

We will learn famous citation data, Cora. The Cora dataset consists of 2708 scientific papers classified into one of seven classes.



In [4]:

```
#load data
path = "/content/drive/MyDrive/data/cora/"
dataset = "cora"

#node data processing
node_data = np.genfromtxt(f"{path}{dataset}.content", dtype=np.dtype(str))
print(f"The number of nodes : {len(node_data)}")
print(node_data)
```

```
The number of nodes : 2708
[['31336' '0' '0' ... '0' '0' 'Neural_Networks']
 ['1061127' '0' '0' ... '0' '0' 'Rule_Learning']
 ['1106406' '0' '0' ... '0' '0' 'Reinforcement_Learning']
 ...
 ['1128978' '0' '0' ... '0' '0' 'Genetic_Algorithms']
 ['117328' '0' '0' ... '0' '0' 'Case_Based']
 ['24043' '0' '0' ... '0' '0' 'Neural_Networks']]
```

In [5]:

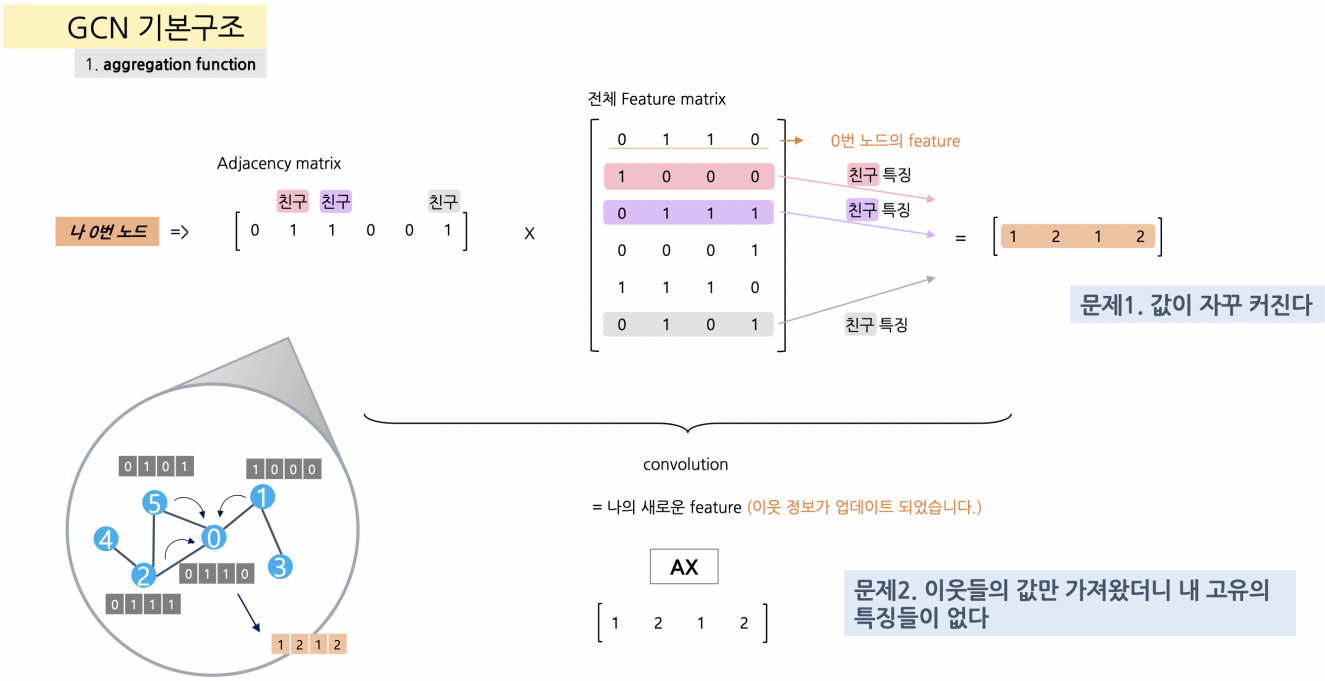
```
idx = np.array(node_data[:,0], dtype=np.int32)
features = sp.csr_matrix(node_data[:,1:-1], dtype=np.float32)
labels = encode_onehot(node_data[:,-1])

idx_map = {j: i for i, j in enumerate(idx)}

print(f'idx : {node_data[12,0]}')
print(f'a feature size : {len(node_data[12,1:-1])}, {node_data[12,1:-1]}')
print(f'a original label : {node_data[12,-1]}')
print(f"encoding label : {labels[12]}")
```

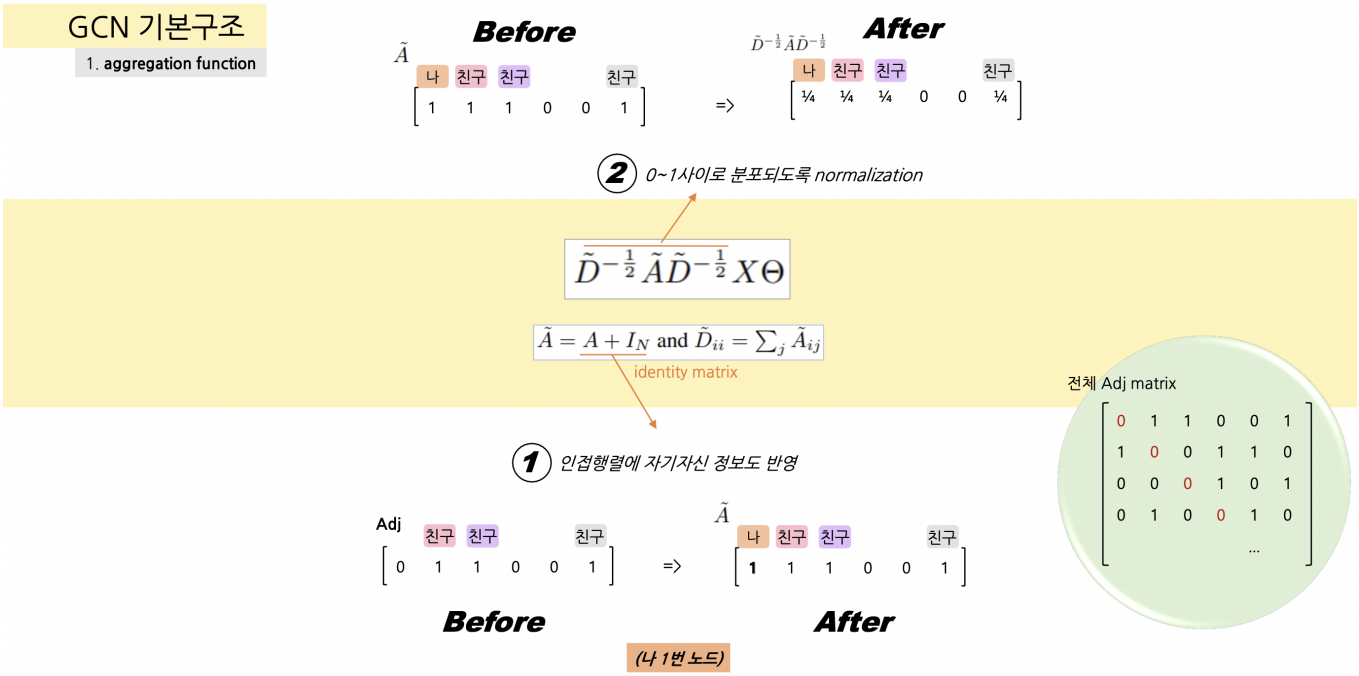
```
idx : 109323
a feature size : 1433, ['0' '0' '1' ... '0' '0' '0']
a original label : Probabilistic_Methods
encoding label : [1 0 0 0 0 0 0]
```

In [6]:

```
#edge data processing
edge_data = np.genfromtxt(f'{path}{dataset}.cites', dtype=np.int32)
edges = np.array(list(map(idx_map.get, edge_data.flatten())), dtype=np.int32)
edges = edges.reshape(edge_data.shape)

print(f'{len(edge_data)} ₩n raw data ₩n {edge_data[:10,]}')
print(f'₩n start node -> end node ₩n {edges}')
```

```
5429
 raw data
 [[     35    1033]
 [     35  103482]
 [     35  103515]
 [     35 1050679]
 [     35 1103960]
 [     35 1103985]
 [     35 1109199]
 [     35 1112911]
 [     35 1113438]
 [     35 1113831]]

 start node -> end node
 [[ 163  402]
 [ 163  659]
 [ 163 1696]
 ...
 [1887 2258]
 [1902 1887]
 [ 837 1686]]
```





Problem 1(20point)

Practice handling an adjacency matrix for graph data

1. Create an adjacency matrix. (5 points)
2. Generate the features matrix for each node. (Number of features should be larger than the number of nodes.) (5 points)
3. Multiply two matrices! (5 points)

친구들의 정보를 반영하면 어떤 장점과 단점이 있을까요? 자유롭게 적어보세요 (5점)

In [7]:

```
#Write your answer here
sample_adj = np.random.random((3,3))
sample_adj = sample_adj > 0.5
sample_adj = sample_adj * 1
np.fill_diagonal(sample_adj,0)
sample_adj
```

Out[7]:

```
array([[0, 0, 1],
       [0, 0, 1],
       [0, 1, 0]])
```

In [8]:

```
sample_features = np.random.randint(10,size=(3,5))
sample_features
```

Out[8]:

```
array([[2, 0, 6, 5, 5],
       [1, 6, 4, 0, 7],
       [7, 7, 5, 1, 1]])
```

In [9]:

```
new_feature = np.matmul(sample_adj, sample_features)
new_feature
#장단점
#장점 나에게 영향력을 주는 정보들을 활용할수 있다
#단점 깊을수록 영향력이 적다, 연산량이 증가한다
```

Out[9]:

```
array([[7, 7, 5, 1, 1],
       [7, 7, 5, 1, 1],
       [1, 6, 4, 0, 7]])
```

In [10]:

```
adj = sp.coo_matrix((np.ones(edges.shape[0]), (edges[:,0], edges[:,1])),
                    shape = (labels.shape[0], labels.shape[0]),
                    dtype = np.float32)
adj = adj + adj.T.multiply(adj.T > adj) - adj.multiply(adj.T > adj)

#normalize
features = normalize(features)
adj = normalize(adj + sp.eye(adj.shape[0]))
```

$$\text{adj} = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} + \text{adj.T} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \text{Adj.T>adj} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} - \text{adj} = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \text{Adj.T>adj} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \qquad = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

In [11]:

```
#data split - semi supervised learning
idx_train = range(140)
idx_val = range(200, 500)
idx_test = range(500, 1500)

#tensorize
features = torch.FloatTensor(np.array(features.todense()))
labels = torch.LongTensor(np.where(labels)[1])
adj = sparse_mx_to_torch_sparse_tensor(adj)

idx_train = torch.LongTensor(idx_train)
idx_val = torch.LongTensor(idx_val)
idx_test = torch.LongTensor(idx_test)
```

Activation function

$$Z = f(X, A) = \mathrm{softmax}\Big(\hat{A}\ \mathrm{ReLU}\Big(\hat{A}XW^{(0)}\Big)\ W^{(1)}\Big)$$

A : NxN matrix
X : NxF matrix
W : F x Output dim matrix

In [28]:

```
# GCN layer
class GraphConvolution(Module):
    def __init__(self, in_features, out_features, bias=True):
        super(GraphConvolution, self).__init__()
        self.in_features = in_features
        self.out_features = out_features
        self.weight = Parameter(torch.FloatTensor(in_features, out_features))
        print(self.weight.shape)
        if bias:
            self.bias = Parameter(torch.FloatTensor(out_features))
        else:
            self.register_parameter('bias', None)
        print("bias is:",self.bias.shape)
        self.reset_parameters()

    def reset_parameters(self):
        stdv = 1. / math.sqrt(self.weight.size(1))
        self.weight.data.uniform_(-stdv, stdv)
        if self.bias is not None:
            self.bias.data.uniform_(-stdv, stdv)

    def forward(self, input, adj):
        support = torch.mm(input, self.weight)
        output = torch.spmm(adj, support)
        if self.bias is not None:
            return output + self.bias
        else:
            return output
```

In [29]:

```
#model
class GCN(nn.Module):
    def __init__(self, nfeat, nhid, nclass, dropout):
        super(GCN, self).__init__()

        self.gc1 = GraphConvolution(nfeat, nhid)
        self.gc2 = GraphConvolution(nhid, nclass)
        self.dropout = dropout

    def forward(self, x, adj):
        x = F.relu(self.gc1(x, adj))
        x = F.dropout(x, self.dropout, training=self.training)
        x = self.gc2(x, adj)
        return F.log_softmax(x, dim=1)
```

In [30]:

```
#Main
import easydict

args = easydict.EasyDict({"no-cuda":False, "fastmode":False, "seed":42, ₩
                          "epochs":200, "lr":0.01, "weight_decay":5e-4, ₩
                          "hidden":16, "dropout":0.5, "cuda":True})

np.random.seed(args.seed)
torch.manual_seed(args.seed)

model = GCN(nfeat=features.shape[1],
            nhid=args.hidden,
            nclass=labels.max().item() + 1,
            dropout=args.dropout)

optimizer = optim.Adam(model.parameters(),
                       lr=args.lr, weight_decay=args.weight_decay)

if args.cuda:
    torch.cuda.manual_seed(args.seed)
    model.cuda()
    features = features.cuda()
    adj = adj.cuda()
    labels = labels.cuda()
    idx_train = idx_train.cuda()
    idx_val = idx_val.cuda()
    idx_test = idx_test.cuda()
```

```
torch.Size([1433, 16])
bias is: torch.Size([16])
torch.Size([16, 7])
bias is: torch.Size([7])
```

In [27]:

```
def train(epoch):
    t = time.time()

    model.train()
    optimizer.zero_grad()
    #forward
    output = model(features, adj)

    #calculate loss
    loss_train = F.nll_loss(output[idx_train], labels[idx_train])
    acc_train = accuracy(output[idx_train], labels[idx_train])

    loss_train.backward()
    optimizer.step()
    #calculate validation loss
    loss_val = F.nll_loss(output[idx_val], labels[idx_val])
    acc_val = accuracy(output[idx_val], labels[idx_val])
    print(f'[Epoch {epoch+1:04d}] Train_loss: {loss_train.item():.4f}, Train_accuracy: {acc_tra
in.item():.4f},',
          f'Val_loss: {loss_val.item():.4f}, Val_accuracy: {acc_val.item():.4f}, Wn #Time: {tim
e.time() - t:.4f}')

def accuracy(output, labels):
    preds = output.max(1)[1].type_as(labels)
    correct = preds.eq(labels).double()
    correct = correct.sum()
    return correct / len(labels)
```

In [ ]:

```
#train model
t_total = time.time()
for epoch in range(100):#args.epochs):
    train(epoch)
print(f"Total time elapsed: {time.time() - t_total:.4f}")
```

In [17]:

```
def test():
    model.eval()
    output = model(features, adj)

    loss_test = F.nll_loss(output[idx_train], labels[idx_train])
    acc_test = accuracy(output[idx_train], labels[idx_train])
    print("Test set results:",
          f"loss= {loss_test.item():.4f}",
          f"accuracy= {acc_test.item():.4f}")
```

In [18]:

```
# Testing
test()
```

```
Test set results: loss= 0.6473 accuracy= 0.9000
```

Problem2(10point)

GCN 모델에서 학습을 하고 값이 바뀌는 부분은 어디일까요? \ 코드를 찾고 weight(learnable parameter)의 크기를 설명해주세요. \ (hint, GCN 모델은 2층입니다)

In [19]:

```
#write your answer
# self.weight = Parameter(torch.FloatTensor(in_features, out_features)) 이부분이 바뀝니다
# shape을 찍어봤을때 위에 결과에 의하면 1433 * 16 + 16 + 16 * 7 = 23056
```

In [ ]:

```
!apt-get install texlive texlive-xetex texlive-latex-extra pandoc
!pip install pypandoc
from google.colab import drive
drive.mount('/content/drive')
!jupyter nbconvert --to html '/content/drive/MyDrive/인공지능프로젝트/week7_lab_final.ipynb'
```

```
Reading package lists... Done
Building dependency tree
Reading state information... Done
pandoc is already the newest version (1.19.2.4~dfsg-1build4).
pandoc set to manually installed.
The following package was automatically installed and is no longer required:
  libnvidia-common-460
Use 'apt autoremove' to remove it.
The following additional packages will be installed:
  fonts-droid-fallback fonts-lato fonts-lmodern fonts-noto-mono fonts-texgyre
  javascript-common libcupsfilters1 libcupsimage2 libgs9 libgs9-common
  libijs-0.35 libjbig2dec0 libjs-jquery libkpathsea6 libpotrace0 libptexenc1
  libruby2.5 libsynctex1 libtexlua52 libtexluajit2 libzzip-0-13 lmodern
  poppler-data preview-latex-style rake ruby ruby-did-you-mean ruby-minitest
  ruby-net-telnet ruby-power-assert ruby-test-unit ruby2.5
  rubygems-integration t1utils tex-common tex-gyre texlive-base
  texlive-binaries texlive-fonts-recommended texlive-latex-base
  texlive-latex-recommended texlive-pictures texlive-plain-generic tipa
Suggested packages:
  fonts-noto apache2 | lighttpd | httpd poppler-utils ghostscript
  fonts-japanese-mincho | fonts-ipafont-mincho fonts-japanese-gothic
  | fonts-ipafont-gothic fonts-arphic-ukai fonts-arphic-uming fonts-nanum ri
  ruby-dev bundler debhelper gv | postscript-viewer perl-tk xpdf-reader
  | pdf-viewer texlive-fonts-recommended-doc texlive-latex-base-doc
  python-pygments icc-profiles libfile-which-perl
  libspreadsheet-parseexcel-perl texlive-latex-extra-doc
  texlive-latex-recommended-doc texlive-pstricks dot2tex prerex ruby-tcltk
  | libtcltk-ruby texlive-pictures-doc vprerex
The following NEW packages will be installed:
  fonts-droid-fallback fonts-lato fonts-lmodern fonts-noto-mono fonts-texgyre
  javascript-common libcupsfilters1 libcupsimage2 libgs9 libgs9-common
  libijs-0.35 libjbig2dec0 libjs-jquery libkpathsea6 libpotrace0 libptexenc1
  libruby2.5 libsynctex1 libtexlua52 libtexluajit2 libzzip-0-13 lmodern
  poppler-data preview-latex-style rake ruby ruby-did-you-mean ruby-minitest
  ruby-net-telnet ruby-power-assert ruby-test-unit ruby2.5
  rubygems-integration t1utils tex-common tex-gyre texlive texlive-base
  texlive-binaries texlive-fonts-recommended texlive-latex-base
  texlive-latex-extra texlive-latex-recommended texlive-pictures
  texlive-plain-generic texlive-xetex tipa
0 upgraded, 47 newly installed, 0 to remove and 12 not upgraded.
Need to get 146 MB of archives.
After this operation, 460 MB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu bionic/main amd64 fonts-droid-fallback all
1:6.0.1r16-1.1 [1,805 kB]
Get:2 http://archive.ubuntu.com/ubuntu bionic/main amd64 fonts-lato all 2.0-2 [2,6
98 kB]
Get:3 http://archive.ubuntu.com/ubuntu bionic/main amd64 poppler-data all 0.4.8-2
[1,479 kB]
Get:4 http://archive.ubuntu.com/ubuntu bionic/main amd64 tex-common all 6.09 [33.0
kB]
Get:5 http://archive.ubuntu.com/ubuntu bionic/main amd64 fonts-lmodern all 2.004.5
-3 [4,551 kB]
Get:6 http://archive.ubuntu.com/ubuntu bionic/main amd64 fonts-noto-mono all 20171
026-2 [75.5 kB]
Get:7 http://archive.ubuntu.com/ubuntu bionic/universe amd64 fonts-texgyre all 201
60520-1 [8,761 kB]
Get:8 http://archive.ubuntu.com/ubuntu bionic/main amd64 javascript-common all 11
[6,066 B]
Get:9 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libcupsfilters1 a
md64 1.20.2-0ubuntu3.1 [108 kB]
Get:10 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libcupsimage2 am
```

```
d64 2.2.7-1ubuntu2.9 [18.6 kB]
Get:11 http://archive.ubuntu.com/ubuntu bionic/main amd64 libijs-0.35 amd64 0.35-1
3 [15.5 kB]
Get:12 http://archive.ubuntu.com/ubuntu bionic/main amd64 libjbig2dec0 amd64 0.13-
6 [55.9 kB]
Get:13 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libgs9-common al
l 9.26~dfsg+0-0ubuntu0.18.04.17 [5,092 kB]
Get:14 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libgs9 amd64 9.2
6~dfsg+0-0ubuntu0.18.04.17 [2,267 kB]
Get:15 http://archive.ubuntu.com/ubuntu bionic/main amd64 libjs-jquery all 3.2.1-1
[152 kB]
Get:16 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libkpathsea6 amd
64 2017.20170613.44572-8ubuntu0.1 [54.9 kB]
Get:17 http://archive.ubuntu.com/ubuntu bionic/main amd64 libpotrace0 amd64 1.14-2
[17.4 kB]
Get:18 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libptexenc1 amd6
4 2017.20170613.44572-8ubuntu0.1 [34.5 kB]
Get:19 http://archive.ubuntu.com/ubuntu bionic/main amd64 rubygems-integration all
1.11 [4,994 B]
Get:20 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 ruby2.5 amd64 2.
5.1-1ubuntu1.12 [48.6 kB]
Get:21 http://archive.ubuntu.com/ubuntu bionic/main amd64 ruby amd64 1:2.5.1 [5,71
2 B]
Get:22 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 rake all 12.3.1-
1ubuntu0.1 [44.9 kB]
Get:23 http://archive.ubuntu.com/ubuntu bionic/main amd64 ruby-did-you-mean all 1.
2.0-2 [9,700 B]
Get:24 http://archive.ubuntu.com/ubuntu bionic/main amd64 ruby-minitest all 5.10.3
-1 [38.6 kB]
Get:25 http://archive.ubuntu.com/ubuntu bionic/main amd64 ruby-net-telnet all 0.1.
1-2 [12.6 kB]
Get:26 http://archive.ubuntu.com/ubuntu bionic/main amd64 ruby-power-assert all 0.
3.0-1 [7,952 B]
Get:27 http://archive.ubuntu.com/ubuntu bionic/main amd64 ruby-test-unit all 3.2.5
-1 [61.1 kB]
Get:28 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libruby2.5 amd64
2.5.1-1ubuntu1.12 [3,073 kB]
Get:29 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libsynctex1 amd6
4 2017.20170613.44572-8ubuntu0.1 [41.4 kB]
Get:30 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libtexlua52 amd6
4 2017.20170613.44572-8ubuntu0.1 [91.2 kB]
Get:31 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libtexluajit2 am
d64 2017.20170613.44572-8ubuntu0.1 [230 kB]
Get:32 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libzzip-0-13 amd
64 0.13.62-3.1ubuntu0.18.04.1 [26.0 kB]
Get:33 http://archive.ubuntu.com/ubuntu bionic/main amd64 lmodern all 2.004.5-3
[9,631 kB]
Get:34 http://archive.ubuntu.com/ubuntu bionic/main amd64 preview-latex-style all
11.91-1ubuntu1 [185 kB]
Get:35 http://archive.ubuntu.com/ubuntu bionic/main amd64 t1utils amd64 1.41-2 [5
6.0 kB]
Get:36 http://archive.ubuntu.com/ubuntu bionic/universe amd64 tex-gyre all 2016052
0-1 [4,998 kB]
Get:37 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 texlive-binaries
amd64 2017.20170613.44572-8ubuntu0.1 [8,179 kB]
Get:38 http://archive.ubuntu.com/ubuntu bionic/main amd64 texlive-base all 2017.20
180305-1 [18.7 MB]
Get:39 http://archive.ubuntu.com/ubuntu bionic/universe amd64 texlive-fonts-recomm
ended all 2017.20180305-1 [5,262 kB]
Get:40 http://archive.ubuntu.com/ubuntu bionic/main amd64 texlive-latex-base all 2
017.20180305-1 [951 kB]
```

```
Get:41 http://archive.ubuntu.com/ubuntu bionic/main amd64 texlive-latex-recommende
d all 2017.20180305-1 [14.9 MB]
Get:42 http://archive.ubuntu.com/ubuntu bionic/universe amd64 texlive all 2017.201
80305-1 [14.4 kB]
Get:43 http://archive.ubuntu.com/ubuntu bionic/universe amd64 texlive-pictures all
2017.20180305-1 [4,026 kB]
Get:44 http://archive.ubuntu.com/ubuntu bionic/universe amd64 texlive-latex-extra
all 2017.20180305-2 [10.6 MB]
Get:45 http://archive.ubuntu.com/ubuntu bionic/universe amd64 texlive-plain-generi
c all 2017.20180305-2 [23.6 MB]
Get:46 http://archive.ubuntu.com/ubuntu bionic/universe amd64 tipa all 2:1.3-20
[2,978 kB]
Get:47 http://archive.ubuntu.com/ubuntu bionic/universe amd64 texlive-xetex all 20
17.20180305-1 [10.7 MB]
Fetched 146 MB in 11s (13.4 MB/s)
Extracting templates from packages: 100%
Preconfiguring packages ...
Selecting previously unselected package fonts-droid-fallback.
(Reading database ... 123934 files and directories currently installed.)
Preparing to unpack .../00-fonts-droid-fallback_1%3a6.0.1r16-1.1_all.deb ...
Unpacking fonts-droid-fallback (1:6.0.1r16-1.1) ...
Selecting previously unselected package fonts-lato.
Preparing to unpack .../01-fonts-lato_2.0-2_all.deb ...
Unpacking fonts-lato (2.0-2) ...
Selecting previously unselected package poppler-data.
Preparing to unpack .../02-poppler-data_0.4.8-2_all.deb ...
Unpacking poppler-data (0.4.8-2) ...
Selecting previously unselected package tex-common.
Preparing to unpack .../03-tex-common_6.09_all.deb ...
Unpacking tex-common (6.09) ...
Selecting previously unselected package fonts-lmodern.
Preparing to unpack .../04-fonts-lmodern_2.004.5-3_all.deb ...
Unpacking fonts-lmodern (2.004.5-3) ...
Selecting previously unselected package fonts-noto-mono.
Preparing to unpack .../05-fonts-noto-mono_20171026-2_all.deb ...
Unpacking fonts-noto-mono (20171026-2) ...
Selecting previously unselected package fonts-texgyre.
Preparing to unpack .../06-fonts-texgyre_20160520-1_all.deb ...
Unpacking fonts-texgyre (20160520-1) ...
Selecting previously unselected package javascript-common.
Preparing to unpack .../07-javascript-common_11_all.deb ...
Unpacking javascript-common (11) ...
Selecting previously unselected package libcupsfilters1:amd64.
Preparing to unpack .../08-libcupsfilters1_1.20.2-0ubuntu3.1_amd64.deb ...
Unpacking libcupsfilters1:amd64 (1.20.2-0ubuntu3.1) ...
```