

Disable glog's "LOG(INFO)" logging

[Ask Question](#)

I'm trying to optimize my c++ program. It uses caffe.

When executing my program, caffe outputs around 1GB (!) of info logs every 15 mins. I suspect this impacts efficiency significantly. But I haven't found how to turn logging off. In [this question](#) someone suggested setting `FLAGS_v` manually.

With the following code I can disable `VLOG` logs by level, but `LOG(x)` logs are unaffected.

First lines in `main()` :

```
FLAGS_v = 1; //disables vlog(2), vlog(3), vlog(4)
VLOG(0) << "Verbose 0";
VLOG(1) << "Verbose 1";
VLOG(2) << "Verbose 2";
VLOG(3) << "Verbose 3";
VLOG(4) << "Verbose 4";
LOG(INFO) << "LOG(INFO)";
LOG(WARNING) << "LOG(WARNING)";
LOG(ERROR) << "LOG(ERROR)";
```

Output:

```
WARNING: Logging before InitGoogleLogging() is written to STDERR
I0523 19:06:51.484634 14115 main.cpp:381] Verbose 0
I0523 19:06:51.484699 14115 main.cpp:382] Verbose 1
I0523 19:06:51.484705 14115 main.cpp:386] LOG(INFO)
W0523 19:06:51.484710 14115 main.cpp:387] LOG(WARNING)
E0523 19:06:51.484715 14115 main.cpp:388] LOG(ERROR)
```

[Home](#)[PUBLIC](#)[Stack Overflow](#)[Tags](#)[Users](#)[Jobs](#)[Teams](#)[Q&A for work](#)

c++

glog

edited Aug 2 '17 at 19:44



Kelsius

143 3 17

asked May 23 '16 at 17:38



jeroen

20 1 5

4 Answers



you need to set your environment variable

5

`GLOG_minloglevel=2`



then run your executable.



You can find more information [here](#) (at the bottom of this page there is a section on stripping `LOG()` s from your code using a macro definition).

edited May 24 '16 at 8:18

answered May 23 '16 at 21:10



Shai

69.3k 22 135 242

1 is there a way to do it from code instead of setting an environment variable? Thanks. – [rkellerm](#) Aug 1 '16 at 8:29



This works in C++ source code.

7

```
google::InitGoogleLogging("XXX");  
google::SetCommandLineOption("GLOG_minloglevel", "2");
```



answered Aug 14 '16 at 2:46



Qi Cai



The environment variable "GLOG_minloglevel" will filter some log but they have been compile in your executable file. If you want to disable them during compiling time, define a macro:



`"#define GOOGLE_STRIP_LOG 1"`

This is the comment in logging.h:

```
111 // The global value of GOOGLE_STRIP_LOG. All the messages logged to
112 // LOG(XXX) with severity less than GOOGLE_STRIP_LOG will not be displayed.
113 // If it can be determined at compile time that the message will not be
114 // printed, the statement will be compiled out.
115 //
116 // Example: to strip out all INFO and WARNING messages, use the value
117 // of 2 below. To make an exception for WARNING messages from a single
118 // file, add "#define GOOGLE_STRIP_LOG 1" to that file _before_ including
119 // base/logging.h
120 #ifndef GOOGLE_STRIP_LOG
121 #define GOOGLE_STRIP_LOG 0
122 #endif
```

edited Jun 15 '17 at 3:30



HDJEMAI

4,280 14 39 56

answered Jun 15 '17 at 2:45



Yu Kunlin

27 4



If you want to turn off log from code level, you can use this.

2

Just add below line in your c++ code at **src/caffe/net.cpp** in `Init` method and build caffe:



```
fLI::FLAGS_minloglevel=3;
```

Partial view of the function where this line should be added:

```
template <typename Dtype>
void Net<Dtype>::Init(const NetParameter& in_param) {
```

```
phase_ = in_param.state().phase();
// Filter layers based on their include/exclude rules and
// the current NetState.
NetParameter filtered_param;
FilterNet(in_param, &filtered_param);
LOG(INFO) << "Initializing net from parameters: " << std::endl
          << filtered_param.DebugString();
// Create a copy of filtered_param with splits added where necessary.
NetParameter param;
InsertSplits(filtered_param, &param);
// Basically, build all the layers and set up their connections.
name_ = param.name();

.
.
.
.
```

Set log level according to your necessity.

[edited Mar 15 '18 at 5:33](#)

answered Jan 18 '18 at 5:35



[MD. Nazmul Kibria](#)

931 8 18