



Άγγελος Λίχας ΑΜ: 19123

Κωνσταντίνος Μητσοκάπας ΑΜ: 19124

Εισαγωγή

Ο αλγόριθμος που κληθήκαμε να υλοποιήσουμε τον αλγόριθμο TextRank για την εξαγωγή λέξεων κλειδιών σε κείμενα γραμμένα στα αγγλικά.

Η γλώσσα που επιλέξαμε για την υλοποίηση μας είναι η Java σε συνδυασμό με την `opennlp`, η οποία βοήθησε στο να ξεχωρίσουμε ποιές λέξεις ήταν “σημαντικές” ώστε να τις χρησιμοποιήσουμε για το δέντρο μας.

Αρχικά υλοποιήσαμε το πρόγραμμα τοπικά στους υπολογιστές μας, σε IDE (Netbeans/IntelliJ), αφού εγκαταστήσαμε `spark`, ώστε να κάνουμε διάφορες δοκιμές. Αργότερα τα περάσαμε όλα στο `vagrant-vm` και προσπαθήσαμε να το τρέξουμε και από εκεί. Χρειάστηκαν να γίνουν κάποιες αλλαγές ως προς το διάβασμα και το γράψιμο των αρχείων οι οποίες έγιναν επιτυχώς, και έτσι το πρόγραμμα τρέχει εν τέλη και μέσω του κατανεμημένου συστήματος και τοπικά.

Arguments και Εντολές εκτέλεσης :

vagrant ssh

hadoop/sbin/start-dfs.sh

spark/sbin/start-all.sh

Arguments: 0 - > input folder

1 - > output folder

2 - > size of Window

3 -> number of iterations

4 - > how many words for ranking

5 -> en-pos-maxent.bin arxeio tagger

Εντολή για υλοποίηση μέσω Vagrant :

```
spark/bin/spark-submit --class com.mycompany.mavenproject1.StopWords  
/vagrant/data/ergasia_spark/mavenproject1/target/mavenproject1-1.0-SNAPSHOT-  
OT-shaded.jar hdfs://localhost:54310/ergasia_spark/input  
hdfs://localhost:54310/ergasia_spark/output 2 4 5  
/vagrant/data/ergasia_spark/en-pos-maxent.bin
```

Εντολη υλοποίησης μέσω IntelliJ/Netbeans terminal :

```
mvn exec:java -Dexec.mainClass=com.mycompany.ergbigdta.StopWords  
-Dexec.args=" /home/angelo/Desktop/BigData/Inspec/docsutf8  
/home/angelo/Desktop/BigData/Kwstas/NetBeansProjects/output 2 3 5  
/home/angelo/Desktop/BigData/Kwstas/NetBeansProjects/en-pos-maxent.bin"
```

Υλοποίηση

Στην αρχή του προγράμματος ξεκινάμε με τις αρχικοποιήσεις των παραμέτρων. Ενώ επίσης αρχικοποιούμε το `javasparkcontext` και το `configuration` του `hdfs` για να μπορέσουμε να διαβάσουμε τους φακέλους και τα αρχεία ενώ αργότερα να μπορέσουμε να γράψουμε τα νέα αρχεία.

ΑΡΧΙΚΟΠΟΙΗΣΕΙΣ

```
int N = Integer.parseInt(args[3]);
int k = Integer.parseInt(args[4]);
SparkConf sparkConf = new
SparkConf().setAppName("StopWords").setMaster("local[2]")
.set("spark.executor.memory", "2g");
JavaSparkContext sc = new JavaSparkContext(sparkConf);

//pernei to configuration instance g to hdfs
Configuration configuration = new Configuration();
//pernei to instance tou HDFS
FileSystem hdfs = FileSystem.get(new URI("hdfs://localhost:54310"),
configuration);
//pernei ta metadata tou fakelu sto args[0]
FileStatus[] fileStatus = hdfs.listStatus(new Path(args[0]));
//xrisimopoiei tin FileUtil gia na parei ta paths ton arxeion
Path[] paths = FileUtil.stat2Paths(fileStatus);
//kai pernei ena ena ta arxeia k ta vazei se mia lista apo paths
List<Path> ListOfFiles = new ArrayList<>();
```

ΔΙΑΒΑΣΜΑ ΓΡΑΜΜΩΝ

```
for (Path file : ListOfFiles ) {
    List<String> myWords = new ArrayList<>();
    //arxikopoiisi listas me lekseis p.x nouns,objects
    List<String> neighbors_List = new ArrayList<>(); // lista pou ta vazei se
tuples
    String fileName = file.getName(); //pernei to onoma tou arxeiou
    JavaRDD<String> lines = sc.textFile(file.toString());
    //pernaei oles tis grammes tou sigkekrimenu file
    lines.cache(); //cache afou tha xrisimopoiithun argotera ksana oi lines
```

ΔΙΑΧΩΡΙΣΜΟΣ ΛΕΞΕΩΝ (ΣΥΝΤΑΚΤΙΚΟ ΦΙΛΤΡΟ)

```
try {
File f = new File(args[5]); //en-pos-maxent.bin gia ton tagger
modelIn = new FileInputStream(f);
POSModel = new POSModel(modelIn);
POSTaggerME tagger = new POSTaggerME(POSModel);
SimpleTokenizer tokenizer = new SimpleTokenizer(); //spazei se lekseis
lines.collect().forEach((line) -> {
    //pernei mia mia tis grammes
    tis spaei se lekseis, kai apofasizei ama tha kratisei tin kathe leksi ksexorista/
    String tokens[] = tokenizer.tokenize(line);
    int flag = 0;
    String[] tagged = tagger.tag(tokens);
    int size = tagged.length;
    for (int i = 0; i < tagged.length; i++) {
        if (tagged[i].equalsIgnoreCase("nn") || tagged[i].equalsIgnoreCase("jj")) {
            //edw koitazume ama i leksi einai noun i adjective
            myWords.add(tokens[i]);
        }
    }
}...
```

Χρησιμοποιώντας το μοντέλο POSModel φτιάξαμε τα κατάλληλα tokens ώστε να τα περάσουμε από το συντακτικό φίλτρο και να λάβουμε μόνο τα ουσιαστικά και τα επίθετα χρησιμοποιώντας το 4ο argument **en-pos-maxent.bin** που χρησιμοποιεί ο συγκεκριμένος tagger της opennlp για να χωρίσει τις λέξεις όπως αναμένεται.

ΕΥΡΕΣΗ ΓΕΙΤΟΝΙΚΩΝ ΚΟΜΒΩΝ

```
for (int i = 0; i < myWords.size(); i++) {
    //koitaei poso makria einai i mia leksi apo tin ali me vasi to megethos tou
    parathiru ,oste na ftiaksei tin lista me tous gitones tis kathe lexis

    for (int l = 1; l < N; l++) {
        if (i + l < myWords.size()) {
            neighbors.add(myWords.get(i) + " " + myWords.get(i + l));
        } else {
            break;
        }
    }
}
```

Με βάση το όρισμα του χρήστη για το αρχικό παράθυρο κάνουμε εύρεση των γειτονικών κόμβων των λέξεων

ΕΥΡΕΣΗ ΚΟΜΒΟΥ ΜΕΓΑΛΥΤΕΡΗΣ ΣΗΜΑΣΙΑΣ

Στο σημείο αυτό βασιστήκαμε στο παράδειγμα του PageRank που υλοποιήθηκε στα πλαίσια του εργαστηρίου του μαθήματος.

```
List<Tuple2<String, Double>> results = new ArrayList<>();
for(String s: neighbors){
    results.add(new Tuple2<>(s, rank/count)); // prosoxi na mhn exw o geitones
}
return results.iterator();
});
// o komvos kai olh h pi8anothta pou erxetai apo tous geitones
JavaPairRDD<String, Double> urlAllContributions =
urlContributions.reduceByKey((a,b)->a+b);
ranks = urlAllContributions.mapValues(s -> 0.15 + s * 0.85);
}
//allazoume tin thesi tis leksis me to rank tis, gia na kanume sort me vasi to key
JavaPairRDD<Double,String> swapped = ranks.flatMapToPair(item ->
Collections.singletonList(item.swap()).iterator());
swapped.collect();
```

ΑΠΟΘΗΚΕΥΣΗ ΣΗΜΑΝΤΙΚΩΝ ΛΕΞΕΩΝ/ΚΟΜΒΩΝ

```
JavaPairRDD<Double,String> swapped = ranks.flatMapToPair(item ->
Collections.singletonList(item.swap()).iterator());
swapped.collect();

JavaPairRDD<Double, String> sorted = swapped.sortByKey(false);
List<String> words = sorted.values().take(k);
//pernume tis k lekseis p thelume
String storepath = args[1]+"/"+fileName; //storepath

Path hdfsWritePath = new Path(storepath);
//pernei to path tu arxeiou sto opoio tha grapsi
FSDataOutputStream fsDataOutputStream = hdfs
.create(hdfsWritePath,true); //to dimiourgei
BufferedWriter bufferedWriter = new BufferedWriter(new
OutputStreamWriter(fsDataOutputStream,StandardCharsets.UTF_8));
//ftiaxnei ton buffer o opoios tha grapsei sto arxio
for(String str : words)
{
    bufferedWriter.write(str);
    //grafei ston buffer mia mia tis lekseis apo to words
    bufferedWriter.newLine();
}
bufferedWriter.close(); //kleinei ton buffer
```

Από τα αποτελέσματα που εξάγονται από την διαδικασία αυτή, ανάλογα με τους κόμβους, παίρνουμε τα εξαγόμενα δεδομένα και αποθηκεύουμε με βάση το όρισμα του αριθμού των λέξεων που έδωσε ο χρήστης, τις λέξεις στα αντίστοιχα αρχεία output, στο path που δίνει ο χρήστης στο 2 argument.

Προβλήματα που αντιμετωπίσαμε

- Κατά την μεταγλώττιση του προγράμματος μέσω του Jar file, μας προέκυψαν διάφορα errors, που εν τέλει ξεπεράστηκαν μετά από έρευνα στο google σε forums με αντίστοιχα προβλήματα, και τροποποίηση του pom.xml ώστε να αναγνωρίζει την κύρια κλάση αλλά και την openHlr με τους taggers/parsers μέσω του vagrant. Έτσι εν τέλει το πρόγραμμα μεταγλωττίζεται μέσω του mavenproject1-1.0-SNAPSHOT-shaded.jar .
- Διάβασμα αρχείων μέσω hdfs path
- Σωστή χρήση της cache() - που όπως είδαμε χρειάζεται όταν κάποια μεταβλητή χρησιμοποιείται σε iterations,for η πολλές φορές ώστε να μην χρειάζεται να ξεφορτώνονται απο την αρχή και έτσι το πρόγραμμα γίνεται ακόμα πιο γρήγορο.

Παραδείγματα εκτέλεσης

Εν τέλει το πρόγραμμα μεταγλωττίζεται μέσω jar file όπως φαίνεται και στα πιο κάτω screenshots.

```
at@at-Inspiron-5593:~/Desktop/lab-spark/data/ergasia_spark/mavenproject1$ mvn package
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.google.inject.internal.cglib.core.$ReflectUtils$1 (file:/usr/share/maven/lib/guice.jar) to method java.lang.ClassLoader.defineClass(java.lang.String,byte[],int,int,java.security.ProtectionDomain)
WARNING: Please consider reporting this to the maintainers of com.google.inject.internal.cglib.core.$ReflectUtils$1
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.mycompany:mavenproject1 >-----
[INFO] Building mavenproject1 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ mavenproject1 ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /home/at/Desktop/lab-spark/data/ergasia_spark/mavenproject1/src/main/resources
[INFO]
[INFO] --- maven-compiler-plugin:2.5.1:compile (default-compile) @ mavenproject1 ---
[INFO] Compiling 1 source file to /home/at/Desktop/lab-spark/data/ergasia_spark/mavenproject1/target/classes
[INFO]
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @ mavenproject1 ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /home/at/Desktop/lab-spark/data/ergasia_spark/mavenproject1/src/test/resources
[INFO]
[INFO] --- maven-compiler-plugin:2.5.1:testCompile (default-testCompile) @ mavenproject1 ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- maven-surefire-plugin:2.12.4:test (default-test) @ mavenproject1 ---
[INFO]
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ mavenproject1 ---
[INFO] Building jar: /home/at/Desktop/lab-spark/data/ergasia_spark/mavenproject1/target/mavenproject1-1.0-SNAPSHOT.jar
```


....

```
[INFO] Attaching shaded artifact.
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 10.204 s
[INFO] Finished at: 2020-06-22T14:39:30+03:00
[INFO] -----
at@at-Inspiron-5593:~/Desktop/lab-spark/data/ergasia_spark/mavenproject1$
```

Παράδειγμα εκτέλεσης 1:

Η συγκεκριμένη εκτέλεση τερμάτισε στα 15 λεπτά, με όλα τα 2000 αρχεία απο το dataset που μας δόθηκε.

Αποτέλεσμα εντολής "hadoop/bin/hadoop fs -ls /ergasia_spark/output" μετά την εκτέλεση. Η εκτέλεση έγινε με 2 iterations, 4 μέγεθος παραθύρου και 5 top words.

```
vagrant@vagrant:~$ hadoop/bin/hadoop fs -ls /ergasia_spark/output
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/
home/vagrant/hadoop-2.6.0/share/hadoop/common/lib/hadoop-auth-2.6.0.jar) to method sun.security.krb5.Conf
ig.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.u
til.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
Found 2000 items
-rw-r--r-- 3 vagrant supergroup 44 2020-06-22 10:23 /ergasia_spark/output/100.txt
-rw-r--r-- 3 vagrant supergroup 44 2020-06-22 10:23 /ergasia_spark/output/1000.txt
-rw-r--r-- 3 vagrant supergroup 41 2020-06-22 10:23 /ergasia_spark/output/1001.txt
-rw-r--r-- 3 vagrant supergroup 46 2020-06-22 10:23 /ergasia_spark/output/1002.txt
-rw-r--r-- 3 vagrant supergroup 45 2020-06-22 10:23 /ergasia_spark/output/1003.txt
-rw-r--r-- 3 vagrant supergroup 38 2020-06-22 10:23 /ergasia_spark/output/1004.txt
-rw-r--r-- 3 vagrant supergroup 42 2020-06-22 10:23 /ergasia_spark/output/1005.txt
-rw-r--r-- 3 vagrant supergroup 38 2020-06-22 10:23 /ergasia_spark/output/1006.txt
-rw-r--r-- 3 vagrant supergroup 34 2020-06-22 10:23 /ergasia_spark/output/1007.txt
-rw-r--r-- 3 vagrant supergroup 47 2020-06-22 10:23 /ergasia_spark/output/1008.txt
-rw-r--r-- 3 vagrant supergroup 44 2020-06-22 10:23 /ergasia_spark/output/1009.txt
-rw-r--r-- 3 vagrant supergroup 40 2020-06-22 10:23 /ergasia_spark/output/1010.txt
-rw-r--r-- 3 vagrant supergroup 41 2020-06-22 10:23 /ergasia_spark/output/1011.txt
-rw-r--r-- 3 vagrant supergroup 43 2020-06-22 10:23 /ergasia_spark/output/1012.txt
-rw-r--r-- 3 vagrant supergroup 41 2020-06-22 10:23 /ergasia_spark/output/1013.txt
-rw-r--r-- 3 vagrant supergroup 47 2020-06-22 10:23 /ergasia_spark/output/1014.txt
-rw-r--r-- 3 vagrant supergroup 43 2020-06-22 10:23 /ergasia_spark/output/1015.txt
-rw-r--r-- 3 vagrant supergroup 45 2020-06-22 10:23 /ergasia_spark/output/1016.txt
```


Αποτέλεσμα εντολής `hadoop/bin/hadoop fs -cat /ergasia_spark/output/999.txt` μετά την εκτέλεση. Ουσιαστικά παρουσιάζονται οι top 5 λέξεις του αρχείου 999.txt

```
-rw-r--r-- 3 vagrant supergroup 27 2020-06-22 10:40 /ergasia_spark/output/996.txt
-rw-r--r-- 3 vagrant supergroup 54 2020-06-22 10:40 /ergasia_spark/output/997.txt
-rw-r--r-- 3 vagrant supergroup 44 2020-06-22 10:40 /ergasia_spark/output/998.txt
-rw-r--r-- 3 vagrant supergroup 40 2020-06-22 10:40 /ergasia_spark/output/999.txt
vagrant@vagrant:~$ hadoop/bin/hadoop fs -cat /ergasia_spark/output/999.txt
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/
home/vagrant/hadoop-2.6.0/share/hadoop/common/lib/hadoop-auth-2.6.0.jar) to method sun.security.krb5.Conf
ig.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.u
til.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
continuity
spike
level
arrival
encoding
vagrant@vagrant:~$
```

Παράδειγμα εκτέλεσης 2:

Η συγκεκριμένη εκτέλεση τερμάτισε στα 30 λεπτά, με όλα τα 2000 αρχεία απο το dataset που μας δόθηκε.

Αποτέλεσμα εντολής `hadoop/bin/hadoop fs -ls /ergasia_spark/output` μετά την εκτέλεση. Η εκτέλεση έγινε με 3 iterations, 6 μέγεθος παραθύρου και 3 top words.

```
vagrant@vagrant:~$ hadoop/bin/hadoop fs -ls /ergasia_spark/output
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/
home/vagrant/hadoop-2.6.0/share/hadoop/common/lib/hadoop-auth-2.6.0.jar) to method sun.security.krb5.Conf
ig.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.u
til.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
Found 2000 items
-rw-r--r-- 3 vagrant supergroup 27 2020-06-22 11:39 /ergasia_spark/output/100.txt
-rw-r--r-- 3 vagrant supergroup 32 2020-06-22 11:39 /ergasia_spark/output/1000.txt
-rw-r--r-- 3 vagrant supergroup 21 2020-06-22 11:39 /ergasia_spark/output/1001.txt
-rw-r--r-- 3 vagrant supergroup 24 2020-06-22 11:39 /ergasia_spark/output/1002.txt
-rw-r--r-- 3 vagrant supergroup 29 2020-06-22 11:39 /ergasia_spark/output/1003.txt
-rw-r--r-- 3 vagrant supergroup 22 2020-06-22 11:39 /ergasia_spark/output/1004.txt
-rw-r--r-- 3 vagrant supergroup 26 2020-06-22 11:39 /ergasia_spark/output/1005.txt
-rw-r--r-- 3 vagrant supergroup 17 2020-06-22 11:39 /ergasia_spark/output/1006.txt
-rw-r--r-- 3 vagrant supergroup 19 2020-06-22 11:39 /ergasia_spark/output/1007.txt
-rw-r--r-- 3 vagrant supergroup 26 2020-06-22 11:39 /ergasia_spark/output/1008.txt
-rw-r--r-- 3 vagrant supergroup 29 2020-06-22 11:39 /ergasia_spark/output/1009.txt
-rw-r--r-- 3 vagrant supergroup 26 2020-06-22 11:39 /ergasia_spark/output/1010.txt
-rw-r--r-- 3 vagrant supergroup 23 2020-06-22 11:39 /ergasia_spark/output/1011.txt
-rw-r--r-- 3 vagrant supergroup 28 2020-06-22 11:39 /ergasia_spark/output/1012.txt
-rw-r--r-- 3 vagrant supergroup 28 2020-06-22 11:39 /ergasia_spark/output/1013.txt
```

Αποτέλεσμα εντολής `hadoop/bin/hadoop fs -cat /ergasia_spark/output/999.txt` μετά την εκτέλεση. Ουσιαστικά παρουσιάζονται οι top 3 λέξεις του αρχείου 999.txt

```
-rw-r--r-- 3 vagrant supergroup 28 2020-06-22 11:57 /ergasia_spark/output/994.txt
-rw-r--r-- 3 vagrant supergroup 26 2020-06-22 11:57 /ergasia_spark/output/995.txt
-rw-r--r-- 3 vagrant supergroup 18 2020-06-22 11:57 /ergasia_spark/output/996.txt
-rw-r--r-- 3 vagrant supergroup 31 2020-06-22 11:57 /ergasia_spark/output/997.txt
-rw-r--r-- 3 vagrant supergroup 27 2020-06-22 11:57 /ergasia_spark/output/998.txt
-rw-r--r-- 3 vagrant supergroup 23 2020-06-22 11:57 /ergasia_spark/output/999.txt
vagrant@vagrant:~$ hadoop/bin/hadoop fs -cat /ergasia_spark/output/999.txt
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/
home/vagrant/hadoop-2.6.0/share/hadoop/common/lib/hadoop-auth-2.6.0.jar) to method sun.security.krb5.Conf
ig.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.u
til.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
level
spike
continuity
vagrant@vagrant:~$
```

Αποτέλεσμα εντολής `hadoop/bin/hadoop fs -cat /ergasia_spark/output/999.txt` μετά την εκτέλεση. Ουσιαστικά παρουσιάζονται οι top 3 λέξεις του αρχείου 109.txt

```
-rw-r--r-- 3 vagrant supergroup 28 2020-06-22 12:28 /ergasia_spark/output/994.txt
-rw-r--r-- 3 vagrant supergroup 26 2020-06-22 12:28 /ergasia_spark/output/995.txt
-rw-r--r-- 3 vagrant supergroup 18 2020-06-22 12:28 /ergasia_spark/output/996.txt
-rw-r--r-- 3 vagrant supergroup 31 2020-06-22 12:28 /ergasia_spark/output/997.txt
-rw-r--r-- 3 vagrant supergroup 27 2020-06-22 12:28 /ergasia_spark/output/998.txt
-rw-r--r-- 3 vagrant supergroup 23 2020-06-22 12:28 /ergasia_spark/output/999.txt
vagrant@vagrant:~$ hadoop/bin/hadoop fs -cat /ergasia_spark/output/109.txt
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/
home/vagrant/hadoop-2.6.0/share/hadoop/common/lib/hadoop-auth-2.6.0.jar) to method sun.security.krb5.Conf
ig.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.u
til.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
entanglement
measure
dense
vagrant@vagrant:~$
```