Assignment 1 Summary

Method Description (what embeddings you used, equations for the distance): In this word analogy task, we were given four words in each line of text and were asked to create a code that could predict the fourth word based on the first three words. To represent words, I used 50-dimensional GloVe vectors 400,000 words. I began by splitting the vocabulary set and mapped words to their GloVe vector representation in the dictionary. I used cosine similarity to calculate the degree of similarity between two embedding vectors for the two words. It is defined as follows:

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

(ref: The cosine similarity algorithm)

where A and B are the dot product (or inner product) of two vectors. Using the given a,b,c, and d words, the approximate embedding for d is determined by:

$$\hat{V_d} = V_c + (V_b - V_a)$$

The output of the top 1 and top 3 accuracy scored based on cosine distance was then computed.

Task designed : We were asked to design a text file with 40 examples for this assignment, but out of curiosity, I produced six text files with different comparisons and similarities. I designed my task by referring to 'The Big Analogy Test series,' ; two of the text task examples have been mentioned in the table below:

Text type	***Text example***
Adjective-Comparative	angry', 'angrier', 'cheap', 'cheaper'
Verb- Verb with ed	'adds', 'added', 'agrees', 'agreed'

Experiment Results:

Text file number	***Text type***	***Text example***	***Accuracy rate***
Text.1	Similarity	'boy', 'girl', 'brother', 'sister'	best_1 accuracy: 0.65 best_3 accuracy: 0.75
Text.2	Similarity	'sister', 'brother', 'girl', 'boy'	best_1 accuracy: 0.65 best_3 accuracy: 0.7
Text.3	Analogy	'acceptable', 'unacceptable', 'fortunate', 'unfortunate'	best_1 accuracy: 0.0 best_3 accuracy: 0.1
Text.4	Analogy	'acceptable', 'unacceptable', 'good', 'bad'	best_1 accuracy: 0.0 best_3 accuracy: 0.05
Text.5	Similarity	angry', 'angrier', 'cheap', 'cheaper'	best_1 accuracy: 0.32 best_3 accuracy: 0.48
Text.6	Similarity	'adds', 'added', 'agrees', 'agreed'	best_1 accuracy: 0.28 best_3 accuracy: 0.52

Observation: Looking at the results we learned that the vector space worked well for text.1 and text.2 because in both the cases we are trying to find the similar words whereas in the other two tasks (text.3 and text.4) we are finding the antonyms therefore in that case the formula used doesn't solve the problem. The outcomes of the designed tasks (text.5 and text.6) were averaged.

In [12]:
```
import numpy as np
from word2vec import *
from nltk.tokenize import sent_tokenize, word_tokenize
import warnings
import nltk
import gensim

# todo:1
class word_obj:
    cosine_sim=-100
    word=None

    def __init__(self, cosine_sim=-100, word='None'):
        self.cosine_sim = cosine_sim
        self.word = word

    def __str__(self):
        return self.word

    def __repr__(self):
        return self.word
```

In [13]:
```
def read_glove_vecs(glove_file):
    with open(glove_file, 'r') as f:
        words = set()
        word_to_vec_map = {}
        for line in f:
            line = line.strip().split()
            curr_word = line[0]
            words.add(curr_word)
            word_to_vec_map[curr_word] = np.array(line[1:], dtype=np.float6
    return word_to_vec_map
```

In [14]:
```
def cosine_similarity(u, v):
    distance = 0.0
    # Compute the dot product between u and v
    dot = np.dot(u,v)
    # Compute the L2 norm of u
    norm_u = np.sqrt(np.sum(u * u))
    # Compute the L2 norm of v
    norm_v = np.sqrt(np.sum(v * v))
    # Compute the cosine similarity defined by formula (1) (≈1 line)
    cosine_similarity = dot / (norm_u * norm_v)
    return cosine_similarity
```

In [15]:
```
def complete_analogy(word_a, word_b, word_c, word_to_vec_map):
    words = word_to_vec_map.keys()
    best_3_words_obj = [word_obj(), word_obj(), word_obj()] # Initialize be

    # convert words to lower case
    word_a, word_b, word_c = word_a.lower(), word_b.lower(), word_c.lower()

    # Get the word embeddings v_a, v_b, v_c and estimated v_d
    e_a, e_b, e_c = word_to_vec_map[word_a], word_to_vec_map[word_b], word_
    e_d_ = e_c + e_b - e_a

    # loop over the whole word vector set
    for w in words:
        # to avoid best_word being one of the input words, pass on them.
        if w in [word_a, word_b, word_c] :
            continue

        # Compute cosine similarity between the vector (e_b - e_a) and the
        cosine_sim = cosine_similarity(e_d_, word_to_vec_map[w])

        # If the cosine_sim is more than the minimum in best_3, add it to b
        if cosine_sim > best_3_words_obj[2].cosine_sim:
            best_3_words_obj.append(word_obj(cosine_sim, w))
            best_3_words_obj = sorted(best_3_words_obj, key=lambda x: x.cos

    return best_3_words_obj
```

In [16]:
```
def analogy_accuracy_rate(filepath):
    file = open(filepath, "r")
    lines = file.readlines()
    total = 0
    best_1 = 0
    best_3 = 0
    for line in lines:
        words = word_tokenize(line)
        if len(words) == 4:
            total += 1
            best_3_words_obj = complete_analogy(words[0], words[1], words[2
            best_3_words = [o.word for o in best_3_words_obj]
            print('given true a,b,c,d -> ', words)
            print('best_3 match for d item -> ', best_3_words)
            if best_3_words[0] == words[3]:
                best_1 += 1
            if words[3] in best_3_words:
                best_3 += 1
    print("best_1 accuracy: ", best_1/total)
    print("best_3 accuracy: ", best_3/total)
```

In [17]:
```
word_to_vec_map = read_glove_vecs('/Users/nehakardam/Documents/UWclasses /E
```

In [23]:
```
len(word_to_vec_map)
```

Out[23]:
```
400000
```

In [7]:
```
analogy_accuracy_rate("/Users/nehakardam/Documents/UWclasses /EE517 NLP/Lab
analogy_accuracy_rate("/Users/nehakardam/Documents/UWclasses /EE517 NLP/Lab
analogy_accuracy_rate("/Users/nehakardam/Documents/UWclasses /EE517 NLP/Lab
analogy_accuracy_rate("/Users/nehakardam/Documents/UWclasses /EE517 NLP/Lab
```

```
given true a,b,c,d ->  ['her', 'his', 'sisters', 'brothers']
best_3 match for d item ->  ['sons', 'brothers', 'fathers']
given true a,b,c,d ->  ['grandmother', 'grandfather', 'mom', 'dad']
best_3 match for d item ->  ['dad', 'guy', 'uncle']
given true a,b,c,d ->  ['bride', 'groom', 'mother', 'father']
best_3 match for d item ->  ['wife', 'friend', 'daughter']
given true a,b,c,d ->  ['aunt', 'uncle', 'grandma', 'grandpa']
best_3 match for d item ->  ['grandpa', 'dad', 'yogi']
given true a,b,c,d ->  ['niece', 'nephew', 'grandaughter', 'grandson']
best_3 match for d item ->  ['grandnephew', 'karatzaferis', 'forefather']
given true a,b,c,d ->  ['queen', 'king', 'she', 'he']
best_3 match for d item ->  ['he', 'him', 'when']
given true a,b,c,d ->  ['grandma', 'grandpa', 'her', 'his']
best_3 match for d item ->  ['his', 'she', 'him']
given true a,b,c,d ->  ['woman', 'man', 'wife', 'husband']
best_3 match for d item ->  ['friend', 'brother', 'son']
given true a,b,c,d ->  ['daughter', 'son', 'woman', 'man']
best_3 match for d item ->  ['man', 'boy', 'old']
given true a,b,c,d ->  ['stepsister', 'stepbrother', 'daughters', 'sons']
```

Designed task

In [8]:
```
analogy_accuracy_rate("/Users/nehakardam/Documents/UWclasses /EE517 NLP/Lab
```

```
given true a,b,c,d ->  ['angry', 'angrier', 'cheap', 'cheaper']
best_3 match for d item ->  ['costlier', 'pricier', 'affordably']
given true a,b,c,d ->  ['clever', 'cleverer', 'coarse', 'coarser']
best_3 match for d item ->  ['coarser', 'undyed', 'lucani']
given true a,b,c,d ->  ['costly', 'costlier', 'cute', 'cuter']
best_3 match for d item ->  ['adorable', 'perky', 'cuter']
given true a,b,c,d ->  ['dense', 'denser', 'dumb', 'dumber']
best_3 match for d item ->  ['dumber', 'conceited', 'egotistical']
given true a,b,c,d ->  ['fierce', 'fiercer', 'handy', 'handier']
best_3 match for d item ->  ['affordably', 'embraceable', 'smarttoaster']
given true a,b,c,d ->  ['happy', 'happier', 'hardy', 'hardier']
best_3 match for d item ->  ['amies', 'fownes', 'farre']
given true a,b,c,d ->  ['harsh', 'harsher', 'healthy', 'healthier']
best_3 match for d item ->  ['healthier', 'quicker', 'balanced']
given true a,b,c,d ->  ['hot', 'hotter', 'huge', 'huger']
best_3 match for d item ->  ['dwarfed', 'enormous', 'magnified']
given true a,b,c,d ->  ['hungry', 'hungrier', 'lazy', 'lazier']
best_3 match for d item ->  ['dreadfully', 'turny', 'agreeably']
given true a,b,c,d ->  ['lengthy', 'lengthier', 'lucky', 'luckier']
best_3 match for d item ->  ['hungrier', 'prettier', 'vanous']
given true a,b,c,d ->  ['mad', 'madder', 'merry', 'merrier']
best_3 match for d item ->  ['bukka', 'hollyhocks', 'delphiniums']
given true a,b,c,d ->  ['mild', 'milder', 'moist', 'moister']
best_3 match for d item ->  ['drier', 'rainforests', 'savanna']
given true a,b,c,d ->  ['nasty', 'nastier', 'neat', 'neater']
best_3 match for d item ->  ['neater', 'sillier', 'crisper']
given true a,b,c,d ->  ['nice', 'nicer', 'noisy', 'noisier']
best_3 match for d item ->  ['noisier', 'unhappier', 'crashworthy']
given true a,b,c,d ->  ['proud', 'prouder', 'pure', 'purer']
best_3 match for d item ->  ['purer', 'vanillin', 'cruder']
given true a,b,c,d ->  ['risky', 'riskier', 'rocky', 'rockier']
best_3 match for d item ->  ['craggy', 'cliffs', 'bottoms']
given true a,b,c,d ->  ['rude', 'ruder', 'sad', 'sadder']
best_3 match for d item ->  ['huber', 'cadafalch', 'škofja']
given true a,b,c,d ->  ['scary', 'scarier', 'sexy', 'sexier']
best_3 match for d item ->  ['sexier', 'cuter', 'prettier']
given true a,b,c,d ->  ['sticky', 'stickier', 'strict', 'stricter']
best_3 match for d item ->  ['strictures', 'restrictive', 'strictest']
given true a,b,c,d ->  ['strong', 'stronger', 'subtle', 'subtler']
best_3 match for d item ->  ['subtly', 'subtler', 'nuances']
given true a,b,c,d ->  ['sunny', 'sunnier', 'tasty', 'tastier']
best_3 match for d item ->  ['tastier', 'appetizing', 'morsels']
given true a,b,c,d ->  ['tiny', 'tinier', 'tricky', 'trickier']
best_3 match for d item ->  ['unserious', 'trickier', 'dificult']
given true a,b,c,d ->  ['ugly', 'uglier', 'vague', 'vaguer']
best_3 match for d item ->  ['comprehensible', 'couched', 'vaguer']
given true a,b,c,d ->  ['vast', 'vaster', 'weak', 'weaker']
best_3 match for d item ->  ['morose', 'sassier', 'revealingly']
given true a,b,c,d ->  ['wealthy', 'wealthier', 'weird', 'weirder']
best_3 match for d item ->  ['oftentimes', 'thankfully', 'hokey']
best_1 accuracy:  0.32
best_3 accuracy:  0.48
```

In [17]:
```
analogy_accuracy_rate("/Users/nehakardam/Documents/UWclasses /EE517 NLP/Lab
```

```
given true a,b,c,d ->  ['calm', 'calmly', 'complete', 'completely']
best_3 match for d item ->  ['nailed', 'inserted', 'completed']
given true a,b,c,d ->  ['apparent', 'apparently', 'slow', 'slowly']
best_3 match for d item ->  ['fast', 'quickly', 'getting']
given true a,b,c,d ->  ['amazing', 'amazingly', 'free', 'freely']
best_3 match for d item ->  ['allowing', 'restricted', 'allow']
given true a,b,c,d ->  ['cheerful', 'cheerfully', 'occasional', 'occasion
ally']
best_3 match for d item ->  ['frequent', 'jokes', 'endless']
given true a,b,c,d ->  ['most', 'mostly', 'fortunate', 'fortunately']
best_3 match for d item ->  ['terrified', 'unlucky', 'frightened']
given true a,b,c,d ->  ['obvious', 'obviously', 'serious', 'seriously']
best_3 match for d item ->  ['concerned', 'seriously', 'worse']
given true a,b,c,d ->  ['possible', 'possibly', 'quiet', 'quietly']
best_3 match for d item ->  ['sleepy', 'deserted', 'tranquil']
given true a,b,c,d ->  ['professional', 'professionally', 'immediate', 'i
mmediately']
best_3 match for d item ->  ['swiftly', 'respond', 'materialized']
given true a,b,c,d ->  ['quick', 'quickly', 'rapid', 'rapidly']
best_3 match for d item ->  ['rapidly', 'gradually', 'continuously']
given true a,b,c,d ->  ['rapid', 'rapidly', 'sudden', 'suddenly']
best_3 match for d item ->  ['suddenly', 'disappear', 'slowly']
given true a,b,c,d ->  ['rare', 'rarely', 'furious', 'furiously']
best_3 match for d item ->  ['frustrated', 'hesitated', 'angry']
given true a,b,c,d ->  ['reluctant', 'reluctantly', 'happy', 'happily']
best_3 match for d item ->  ['glad', "'m", 'i']
given true a,b,c,d ->  ['safe', 'safely', 'precise', 'precisely']
best_3 match for d item ->  ['measurements', 'accurately', 'trajectorie
s']
given true a,b,c,d ->  ['serious', 'seriously', 'rare', 'rarely']
best_3 match for d item ->  ['critically', 'alive', 'found']
given true a,b,c,d ->  ['sudden', 'suddenly', 'complete', 'completely']
best_3 match for d item ->  ['entire', 'fully', 'then']
given true a,b,c,d ->  ['swift', 'swiftly', 'precise', 'precisely']
best_3 match for d item ->  ['precisely', 'correctly', 'accurately']
given true a,b,c,d ->  ['typical', 'typically', 'reluctant', 'reluctantl
y']
best_3 match for d item ->  ['unwilling', 'persuade', 'convince']
given true a,b,c,d ->  ['usual', 'usually', 'calm', 'calmly']
best_3 match for d item ->  ['remain', 'very', 'leave']
given true a,b,c,d ->  ['swift', 'swiftly', 'cheerful', 'cheerfully']
best_3 match for d item ->  ['behaving', 'perpetually', 'behaved']
given true a,b,c,d ->  ['amazing', 'amazingly', 'reluctant', 'reluctantl
y']
best_3 match for d item ->  ['hesitant', 'unwilling', 'wary']
best_1 accuracy:  0.15
best_3 accuracy:  0.2
```

In [19]:
```
analogy_accuracy_rate("/Users/nehakardam/Documents/UWclasses /EE517 NLP/Lab
```

```
given true a,b,c,d ->  ['acceptable', 'unacceptable', 'competitive', 'unc
ompetitive']
best_3 match for d item ->  ['competition', 'unfair', 'aggressive']
given true a,b,c,d ->  ['aware', 'unaware', 'certain', 'uncertain']
best_3 match for d item ->  ['any', 'presumably', 'specific']
given true a,b,c,d ->  ['reasonable', 'unreasonable', 'productive', 'unpr
oductive']
best_3 match for d item ->  ['inefficient', 'exploitative', 'exploitive']
given true a,b,c,d ->  ['certain', 'uncertain', 'consistent', 'inconsiste
nt']
best_3 match for d item ->  ['optimistic', 'pessimistic', 'remarkably']
given true a,b,c,d ->  ['clear', 'unclear', 'logical', 'illogical']
best_3 match for d item ->  ['logically', 'simplest', 'causal']
given true a,b,c,d ->  ['competitive', 'uncompetitive', 'informative', 'u
ninformative']
best_3 match for d item ->  ['well-done', 'unobjectionable', 'well-resear
ched']
given true a,b,c,d ->  ['possibly', 'impossibly', 'efficient', 'inefficie
nt']
best_3 match for d item ->  ['aerodynamically', 'uncluttered', 'nimble']
given true a,b,c,d ->  ['convincing', 'unconvincing', 'productive', 'unpr
oductive']
best_3 match for d item ->  ['unproductive', 'conflictual', 'unspectacula
r']
given true a,b,c,d ->  ['informative', 'uninformative', 'productive', 'un
productive']
best_3 match for d item ->  ['unharvested', 'unproductive', 'disadvantage
ous']
given true a,b,c,d ->  ['decided', 'undecided', 'responsible', 'irrespons
ible']
best_3 match for d item ->  ['unaligned', 'ideologically', 'respondents']
given true a,b,c,d ->  ['efficient', 'inefficient', 'certain', 'uncertai
n']
best_3 match for d item ->  ['instances', 'dealt', 'exceptions']
given true a,b,c,d ->  ['fortunate', 'unfortunate', 'rational', 'irration
al']
best_3 match for d item ->  ['relation', 'context', 'logical']
given true a,b,c,d ->  ['honest', 'dishonest', 'aware', 'unaware']
best_3 match for d item ->  ['overreacting', 'condoned', 'deterred']
given true a,b,c,d ->  ['impressive', 'unimpressive', 'acceptable', 'unac
ceptable']
best_3 match for d item ->  ['disadvantageous', 'preferable', 'tenable']
given true a,b,c,d ->  ['informative', 'uninformative', 'certain', 'uncer
tain']
best_3 match for d item ->  ['induce', 'tolerate', 'residual']
given true a,b,c,d ->  ['informed', 'uninformed', 'certain', 'uncertain']
best_3 match for d item ->  ['conversely', 'propensity', 'deviations']
given true a,b,c,d ->  ['known', 'unknown', 'impressive', 'unimpressive']
best_3 match for d item ->  ['astonishing', 'surprising', 'remarkable']
given true a,b,c,d ->  ['logical', 'illogical', 'ethical', 'unethical']
best_3 match for d item ->  ['flouting', 'immorality', 'disregarding']
given true a,b,c,d ->  ['possible', 'impossible', 'productive', 'unproduc
tive']
best_3 match for d item ->  ['self-sufficient', 'tremendously', 'immensel
y']
```

```
given true a,b,c,d ->  ['rational', 'irrational', 'tasteful', 'distastefu
l']
best_3 match for d item ->  ['tacky', 'kitschy', 'imitations']
given true a,b,c,d ->  ['sure', 'unsure', 'known', 'unknown']
best_3 match for d item ->  ['originated', 'extinct', 'unknown']
given true a,b,c,d ->  ['Decided', 'undecided', 'logical', 'illogical']
best_3 match for d item ->  ['plausible', 'cogent', '5-to-1']
best_1 accuracy:  0.045454545454545456
best_3 accuracy:  0.13636363636363635
```

In [16]:
```
analogy_accuracy_rate("/Users/nehakardam/Documents/UWclasses /EE517 NLP/Lab
```

```
given true a,b,c,d ->  ['adds', 'added', 'agrees', 'agreed']
best_3 match for d item ->  ['agreed', 'decided', 'intends']
given true a,b,c,d ->  ['allows', 'allowed', 'announces', 'announced']
best_3 match for d item ->  ['announcing', 'announced', 'last']
given true a,b,c,d ->  ['appears', 'appeared', 'applies', 'applied']
best_3 match for d item ->  ['rules', 'laws', 'apply']
given true a,b,c,d ->  ['appoints', 'appointed', 'asks', 'asked']
best_3 match for d item ->  ['he', 'father', 'she']
given true a,b,c,d ->  ['becomes', 'became', 'believes', 'believed']
best_3 match for d item ->  ['recently', 'joined', 'had']
given true a,b,c,d ->  ['considers', 'considered', 'consists', 'consiste
d']
best_3 match for d item ->  ['consisting', 'consist', 'consisted']
given true a,b,c,d ->  ['contains', 'contained', 'continues', 'continue
d']
best_3 match for d item ->  ['continuing', 'continue', 'continued']
given true a,b,c,d ->  ['creates', 'created', 'decides', 'decided']
best_3 match for d item ->  ['decided', 'persuaded', 'chose']
given true a,b,c,d ->  ['describes', 'described', 'develops', 'develope
d']
best_3 match for d item ->  ['develop', 'developed', 'focused']
given true a,b,c,d ->  ['establishes', 'established', 'expects', 'expecte
d']
best_3 match for d item ->  ['year', 'expected', 'recently']
given true a,b,c,d ->  ['fails', 'failed', 'follows', 'followed']
best_3 match for d item ->  ['followed', 'following', 'early']
given true a,b,c,d ->  ['happens', 'happened', 'hears', 'heard']
best_3 match for d item ->  ['heard', 'screams', 'hear']
given true a,b,c,d ->  ['includes', 'included', 'intends', 'intended']
best_3 match for d item ->  ['persuaded', 'intention', 'hoped']
given true a,b,c,d ->  ['introduces', 'introduced', 'involves', 'involve
d']
best_3 match for d item ->  ['used', 'similar', 'which']
given true a,b,c,d ->  ['locates', 'located', 'loses', 'lost']
best_3 match for d item ->  ['is', 'now', 'portion']
given true a,b,c,d ->  ['manages', 'managed', 'marries', 'married']
best_3 match for d item ->  ['marrying', 'cousin', 'father']
given true a,b,c,d ->  ['occurs', 'occurred', 'operates', 'operated']
best_3 match for d item ->  ['offices', 'headquarters', 'depot']
given true a,b,c,d ->  ['performs', 'performed', 'proposes', 'proposed']
best_3 match for d item ->  ['proposed', 'plan', 'proposal']
given true a,b,c,d ->  ['provides', 'provided', 'publishes', 'published']
best_3 match for d item ->  ['publications', 'journals', 'published']
given true a,b,c,d ->  ['receives', 'received', 'refers', 'referred']
best_3 match for d item ->  ['referred', '—', 'latter']
given true a,b,c,d ->  ['relates', 'related', 'remains', 'remained']
best_3 match for d item ->  ['already', 'remain', 'been']
given true a,b,c,d ->  ['replaces', 'replaced', 'represents', 'represente
d']
best_3 match for d item ->  ['which', 'the', 'of']
given true a,b,c,d ->  ['requires', 'required', 'seems', 'seemed']
best_3 match for d item ->  ['probably', 'indeed', 'though']
given true a,b,c,d ->  ['sends', 'sent', 'spends', 'spent']
best_3 match for d item ->  ['spent', 'years', 'months']
given true a,b,c,d ->  ['suggests', 'suggested', 'tells', 'told']
```

```
best_3 match for d item ->  ['asks', 'asked', 'friend']
best_1 accuracy:  0.28
best_3 accuracy:  0.52
```

In [11]:
```
analogy_accuracy_rate("/Users/nehakardam/Documents/UWclasses /EE517 NLP/Lab
```

```
given true a,b,c,d ->  ['aristotle', 'greek', 'balzac', 'french']
best_3 match for d item ->  ['spanish', 'french', 'portuguese']
given true a,b,c,d ->  ['beethoven', 'german', 'caesar', 'roman']
best_3 match for d item ->  ['mercenaries', 'forces', 'foreign']
given true a,b,c,d ->  ['confucius', 'chinese', 'copernicus', 'polish']
best_3 match for d item ->  ['russian', 'foreign', 'currency']
given true a,b,c,d ->  ['darwin', 'english', 'depp', 'american']
best_3 match for d item ->  ['plays', 'actor', 'comedy']
given true a,b,c,d ->  ['descartes', 'french', 'dickens', 'english']
best_3 match for d item ->  ['british', 'american', 'britain']
given true a,b,c,d ->  ['dostoyevsky', 'russian', 'edison', 'american']
best_3 match for d item ->  ['union', 'motors', 'ukrainian']
given true a,b,c,d ->  ['einstein', 'jewish', 'euclid', 'greek']
best_3 match for d item ->  ['netzarim', 'synagogues', 'settlements']
given true a,b,c,d ->  ['fermi', 'italian', 'galilei', 'italian']
best_3 match for d item ->  ['italy', 'nun', 'francesco']
given true a,b,c,d ->  ['gorbachev', 'russian', 'hawking', 'english']
best_3 match for d item ->  ['bulk', 'specialized', 'sophisticated']
given true a,b,c,d ->  ['hegel', 'german', 'hitler', 'german']
best_3 match for d item ->  ['nazi', 'russian', 'polish']
given true a,b,c,d ->  ['homer', 'greek', 'hume', 'scottish']
best_3 match for d item ->  ['cypriot', 'pākehā', 'koine']
given true a,b,c,d ->  ['jolie', 'america', 'kant', 'german']
best_3 match for d item ->  ['socialism', 'fundamental', 'philosophy']
given true a,b,c,d ->  ['kepler', 'german', 'lavoisier', 'french']
best_3 match for d item ->  ['polish', 'french', 'germans']
given true a,b,c,d ->  ['leibniz', 'german', 'lenin', 'russian']
best_3 match for d item ->  ['russian', 'soviet', 'russia']
given true a,b,c,d ->  ['lennon', 'english', 'lincoln', 'american']
best_3 match for d item ->  ['class', 'served', '—']
given true a,b,c,d ->  ['locke', 'english', 'machiavelli', 'italian']
best_3 match for d item ->  ['translations', 'colloquial', 'translated']
given true a,b,c,d ->  ['marx', 'german', 'maxwell', 'scottish']
best_3 match for d item ->  ['british', 'belgian', 'canadian']
given true a,b,c,d ->  ['mencius', 'chinese', 'Michelangelo', 'italian']
best_3 match for d item ->  ['as', 'also', 'well']
given true a,b,c,d ->  ['mozart', 'german', 'napoleon', 'french']
best_3 match for d item ->  ['germans', 'soviet', 'soviets']
given true a,b,c,d ->  ['newton', 'english', 'pascal', 'french']
best_3 match for d item ->  ['portuguese', 'italian', 'spanish']
given true a,b,c,d ->  ['plato', 'greek', 'raphael', 'italian']
best_3 match for d item ->  ['dutch', 'eurozone', 'french']
given true a,b,c,d ->  ['rousseau', 'french', 'spinoza', 'dutch']
best_3 match for d item ->  ['italian', 'dutch', 'spanish']
given true a,b,c,d ->  ['stalin', 'soviet', 'strauss', 'austrian']
best_3 match for d item ->  ['european', 'u.k.', 'europe']
given true a,b,c,d ->  ['tchaikovsky', 'russian', 'tolstoi', 'russian']
best_3 match for d item ->  ['afghan', 'enclave', 'tajik']
given true a,b,c,d ->  ['truman', 'american', 'wagner', 'german']
best_3 match for d item ->  ['trio', 'group', 'duo']
best_1 accuracy:  0.04
best_3 accuracy:  0.16
```

In [10]:
```
analogy_accuracy_rate("/Users/nehakardam/Documents/UWclasses /EE517 NLP/Lab
```

```
given true a,b,c,d ->  ['ant', 'black', 'apple', 'red']
best_3 match for d item ->  ['orange', 'brand', 'white']
given true a,b,c,d ->  ['blackboard', 'black', 'blood', 'red']
best_3 match for d item ->  ['causes', 'and', 'face']
given true a,b,c,d ->  ['blueberry', 'blue', 'broccoli', 'green']
best_3 match for d item ->  ['red', 'trimmed', 'brown']
given true a,b,c,d ->  ['bruise', 'blue', 'cabbage', 'green']
best_3 match for d item ->  ['orange', 'apples', 'olive']
given true a,b,c,d ->  ['carrot', 'orange', 'cauliflower', 'green']
best_3 match for d item ->  ['purple', 'florets', 'pink']
given true a,b,c,d ->  ['celery', 'green', 'cherry', 'red']
best_3 match for d item ->  ['white', 'gray', 'jackson']
given true a,b,c,d ->  ['chocolate', 'brown', 'cloud', 'white']
best_3 match for d item ->  ['guard', 'foley', 'reserve']
given true a,b,c,d ->  ['coal', 'black', 'coffee', 'black']
best_3 match for d item ->  ['shirt', 'shirts', 'dress']
given true a,b,c,d ->  ['cranberry', 'red', 'cream', 'white']
best_3 match for d item ->  ['blue', 'black', 'yellow']
given true a,b,c,d ->  ['crow', 'black', 'cucumber', 'green']
best_3 match for d item ->  ['chopped', 'onion', 'diced']
given true a,b,c,d ->  ['emerald', 'green', 'fridge', 'whitek']
best_3 match for d item ->  ['filling', 'fill', 'refrigerator']
given true a,b,c,d ->  ['frog', 'green', 'grapes', 'black']
best_3 match for d item ->  ['olive', 'wines', 'sauvignon']
given true a,b,c,d ->  ['grass', 'green', 'leaves', 'green']
best_3 match for d item ->  ['brown', 'white', 'bright']
given true a,b,c,d ->  ['milk', 'white', 'paper', 'white']
best_3 match for d item ->  ['pointed', 'framed', 'gray']
given true a,b,c,d ->  ['parsley', 'green', 'peony', 'red']
best_3 match for d item ->  ['mural', 'disneyland', 'exhibit']
given true a,b,c,d ->  ['pepper', 'black', 'potato', 'brown']
best_3 match for d item ->  ['traditional', 'resemble', 'popular']
given true a,b,c,d ->  ['radish', 'red', 'raven', 'black']
best_3 match for d item ->  ['golden', 'rainbow', 'jackson']
given true a,b,c,d ->  ['rose', 'red', 'ruby', 'red']
best_3 match for d item ->  ['purple', 'pink', 'buckskin']
given true a,b,c,d ->  ['salt', 'white', 'sapphire', 'blue']
best_3 match for d item ->  ['ruby', 'crouching', 'striped']
given true a,b,c,d ->  ['sea', 'blue', 'sky', 'blue']
best_3 match for d item ->  ['shirt', 'bright', 'pink']
given true a,b,c,d ->  ['snow', 'white', 'soil', 'black']
best_3 match for d item ->  ['protected', 'black', 'green']
given true a,b,c,d ->  ['spinach', 'green', 'sugar', 'white']
best_3 match for d item ->  ['blue', 'black', 'white']
given true a,b,c,d ->  ['sun', 'yellow', 'swan', 'white']
best_3 match for d item ->  ['otter', 'goose', 'mink']
given true a,b,c,d ->  ['tea', 'black', 'tomato', 'red']
best_3 match for d item ->  ['red', 'collar', 'peppers']
given true a,b,c,d ->  ['toothpaste', 'white', 'yoghurt', 'white']
best_3 match for d item ->  ['black', 'green', 'red']
best_1 accuracy:  0.04
best_3 accuracy:  0.12
```

In [ ]: