



Reading Thieves' Cant: Automatically Identifying and Understanding Dark Jargons from Cybercrime Marketplaces

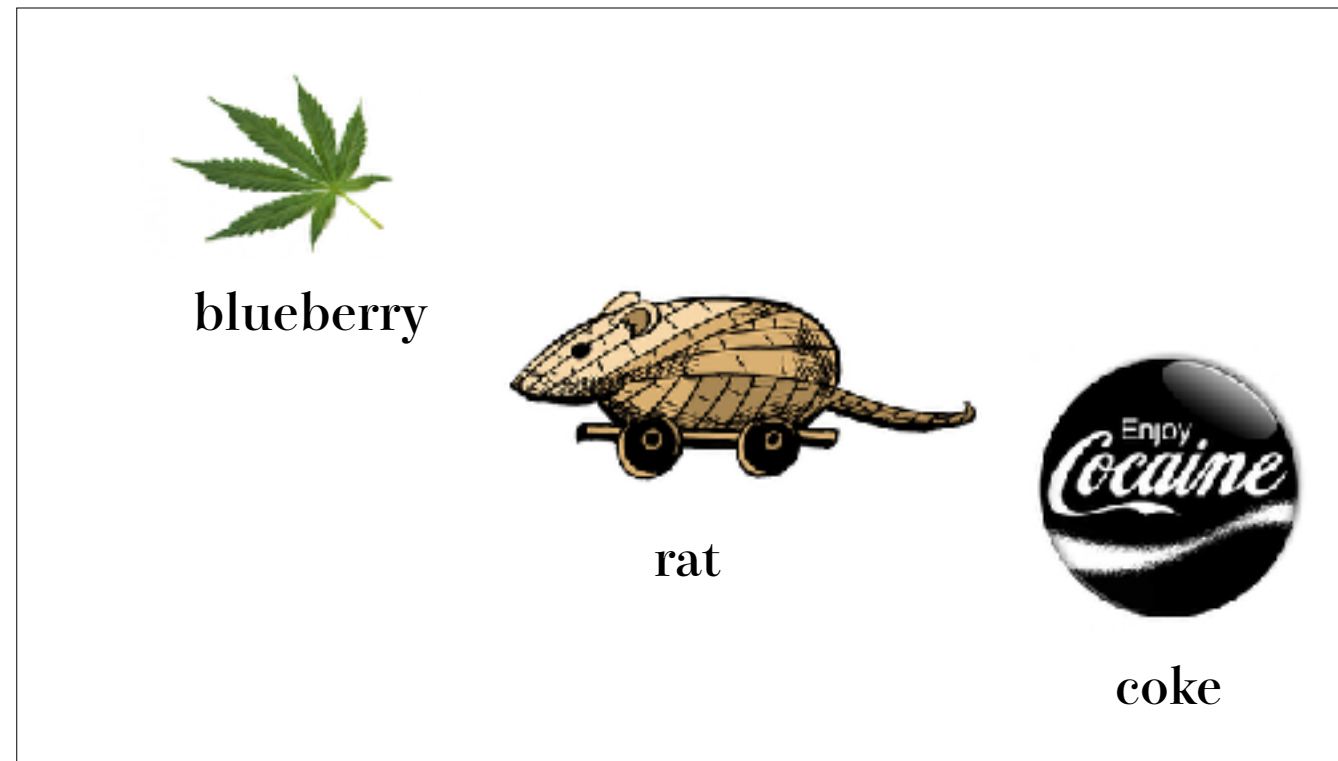
Kan Yuan, Haoran Lu, Xiaojing Liao, and XiaoFeng Wang
Indiana University Bloomington





blueberry

The second examples



One more example.

Rat, also known as remote access trojan. You must be

Words like rat, blueberry and coke are jargons. They have their ordinary meanings, but they are used differently by a particular group



Words like rat, blueberry and coke, that have the ordinary meanings, while are used differently by s particular profession or group are called jargons.

In fact Jargons are extensively used in the underground forums by cyber-criminals for a variety of reasons.

It has become an obstacle

Such deceptive content makes underground communication less conspicuous and difficult to detect, and in some cases, even allows the criminals to communicate through public forums. Hence, automatic discovery and understanding of these dark jargons are highly valuable for understanding various cybercrime activities and mitigating the threats they pose.



CANTREADER

Cantreader an unsupervised approach to automatically detect and understand dark jargon

Let's start with the detection

Key Idea

context = semantics

Key idea is simple, we are going to look into the semantics.

Because communication traces from dark forums are partially obfuscated

Where the key words are replaced with jargons.

Although the jargons themselves are hard to deal with directly. So we can still investigate the context to find the clues of jargons

My fav is slayers new **rat**, its
open source, gonna have his
rootkit implemented into it.

Rat has been used as working
animal. Tasks for working rats
include the sniffing of gunpowder
residue, demining, acting and
animal-assisted therapy.

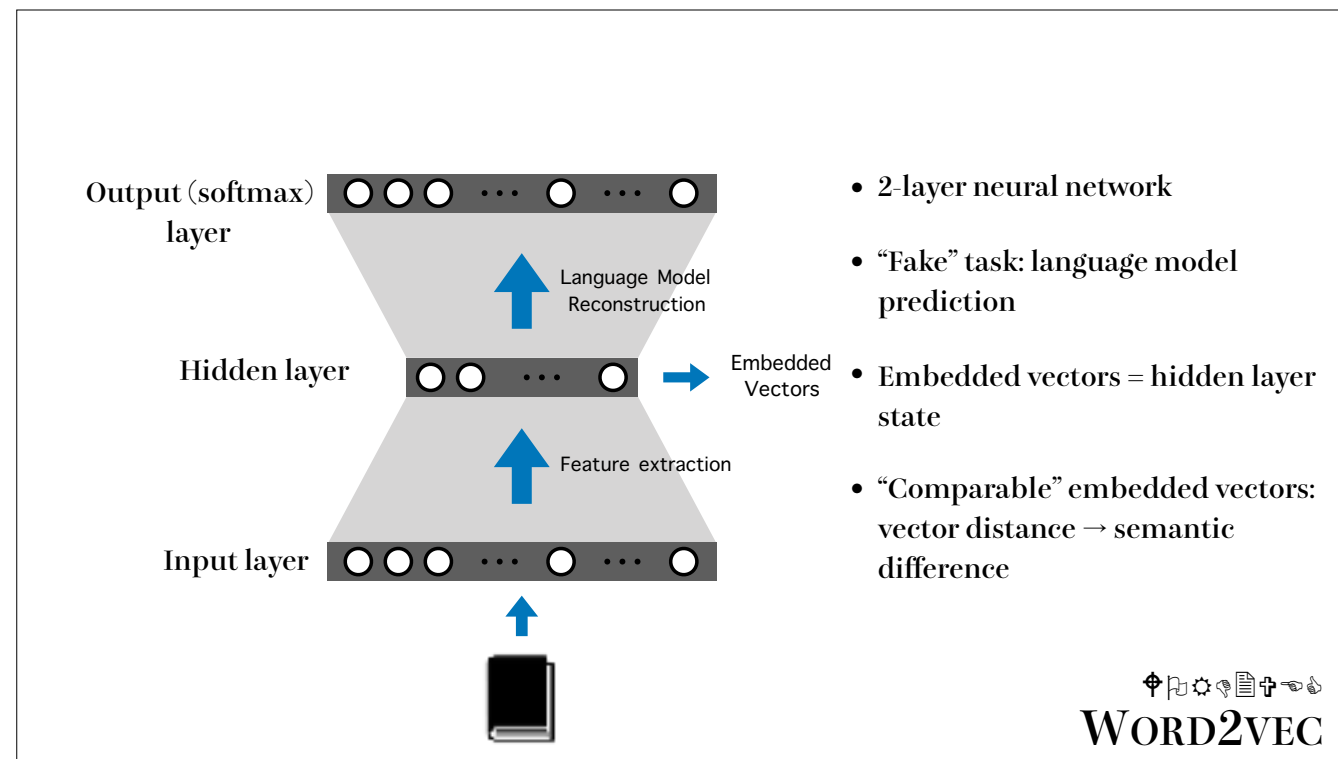
Let's look at the two pieces of text, both using the jargon word rat, but with different meaning

Rat used as jargon, context: opensource, rootkit, slayers, implement

Rat means mouse, context: animal, working, therapy

Therefore, if a word is used as a jargon in an underground forum, its context in that forum ought to be totally different from that in the legit communication traces. There is how we are going to detect dark jargons.

To better extract a word context information, and **directly use that information in the semantic comparison**,



Word2vec (Tomas Mikolov 2013) is a word embedding technique

it use a 2-layer shallow NN

Fake task: language model prediction, Language model: predict the context of given word

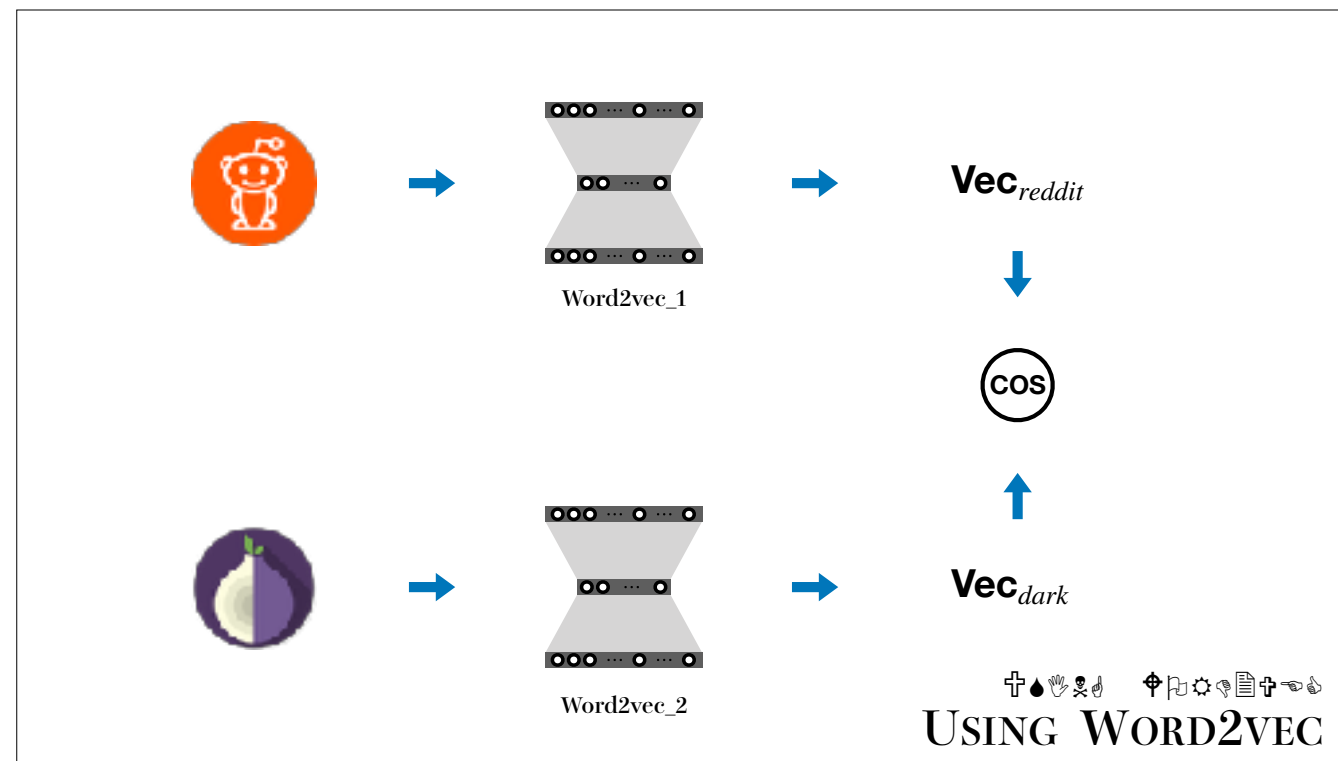
Idea is like auto-encoder, 1st layer extract features, 2nd layer reconstruction

After training, 2nd layer ignored,

the embedded vectors are not just the densened feature vectors of the words, they **actually represent the semantics** of the words in the numeric form. So it shows some interesting property:

Comparable: we say two vectors are comparable, means we can use the distance of embedded vectors to estimate words' Semantic difference

With this property, it seems that we are already ready to find dark jargons



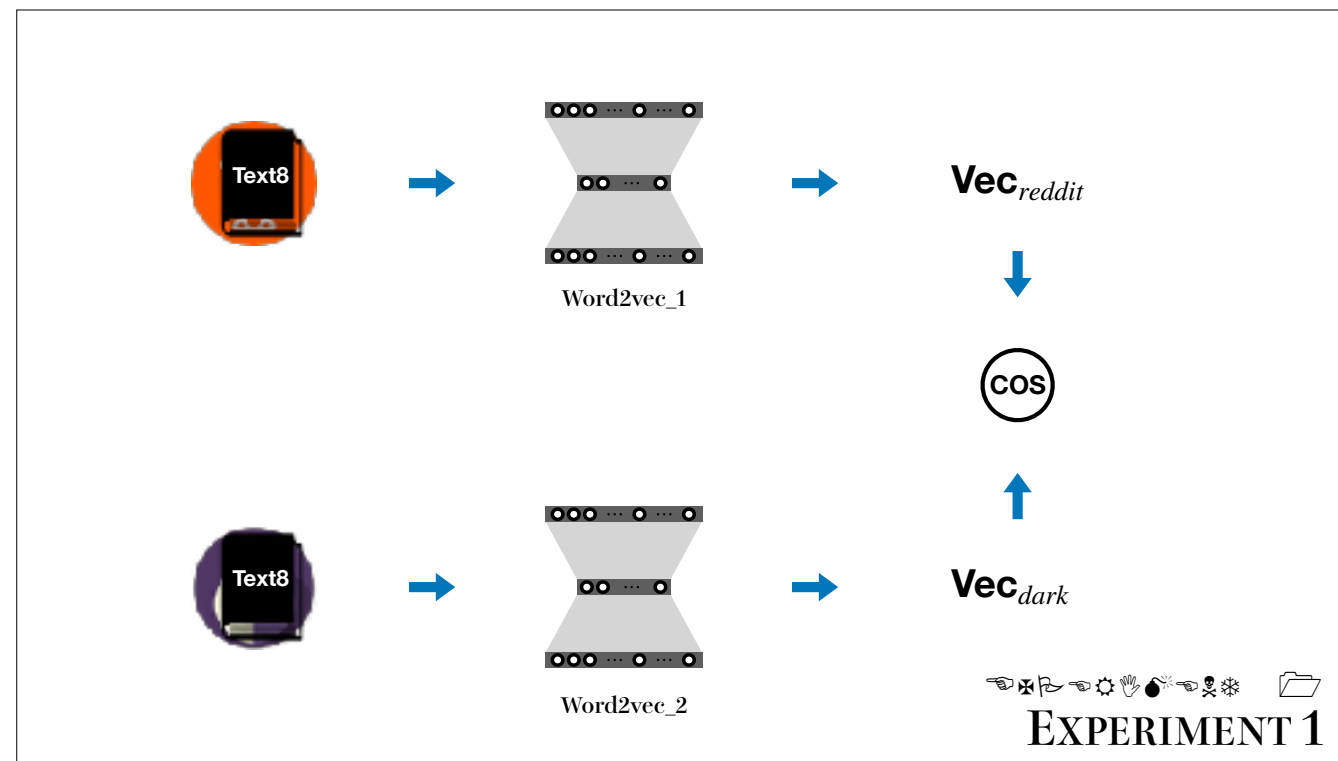
This property sounds very promising, it seems that we are already ready to find dark jargons!

The idea is differential analysis.

Two corpora, Ordinary forum and underground forum

Each word has two vectors, comparing the vectors to see if the word has different contexts/meaning between the two corpora.

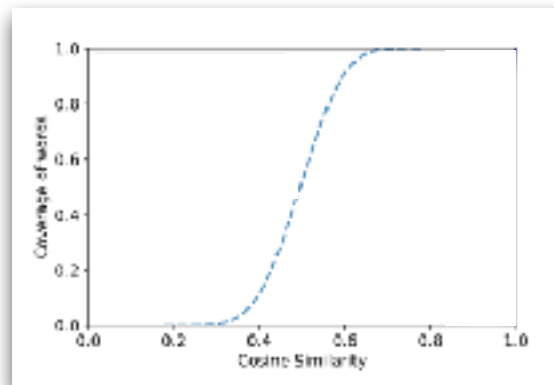
Problem, are vectors from tow separately trained models comparable



We investigate this with an experiment

Since we use the same corpus, the context of the word should be same
If truly comparable, cosine similarity of the vectors should be close 1.

Cross-corpus Semantic Comparison: Word2vec

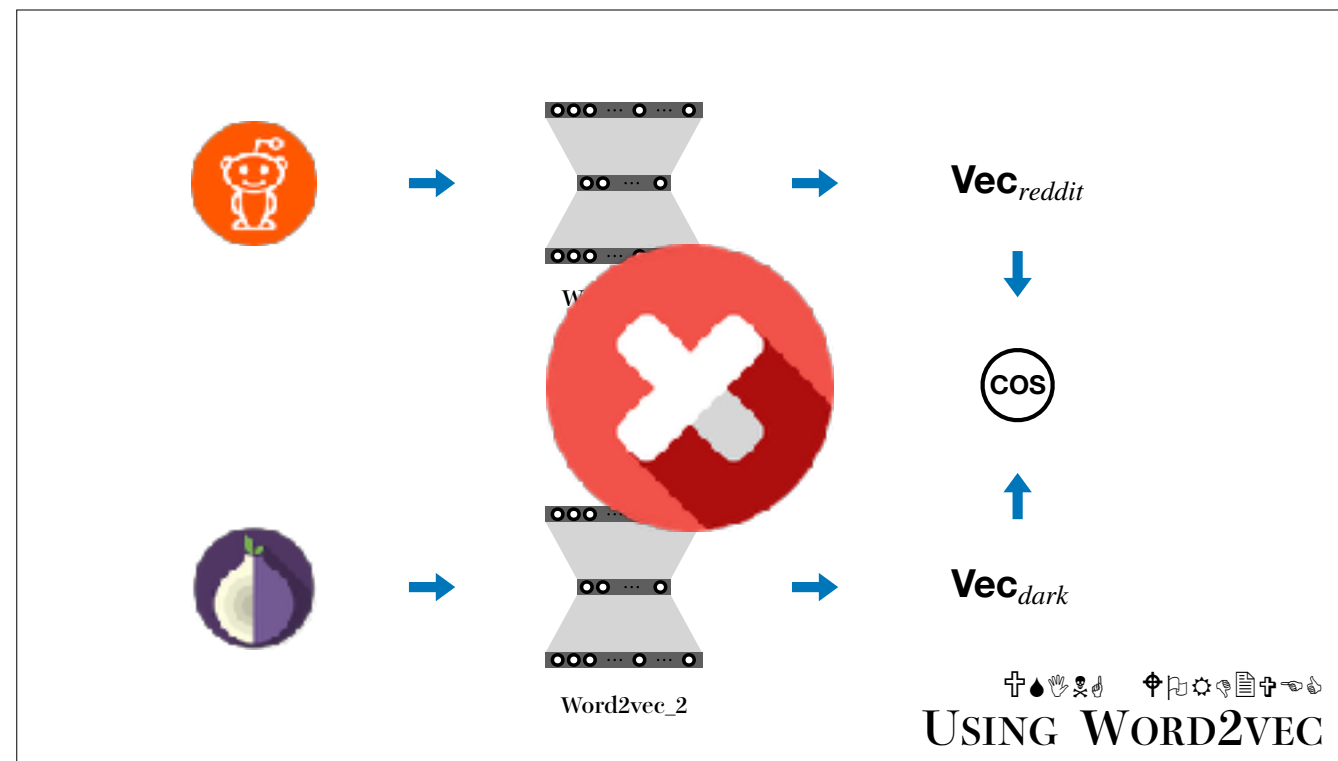


$$\mu = 0.49, \sigma = 0.078$$

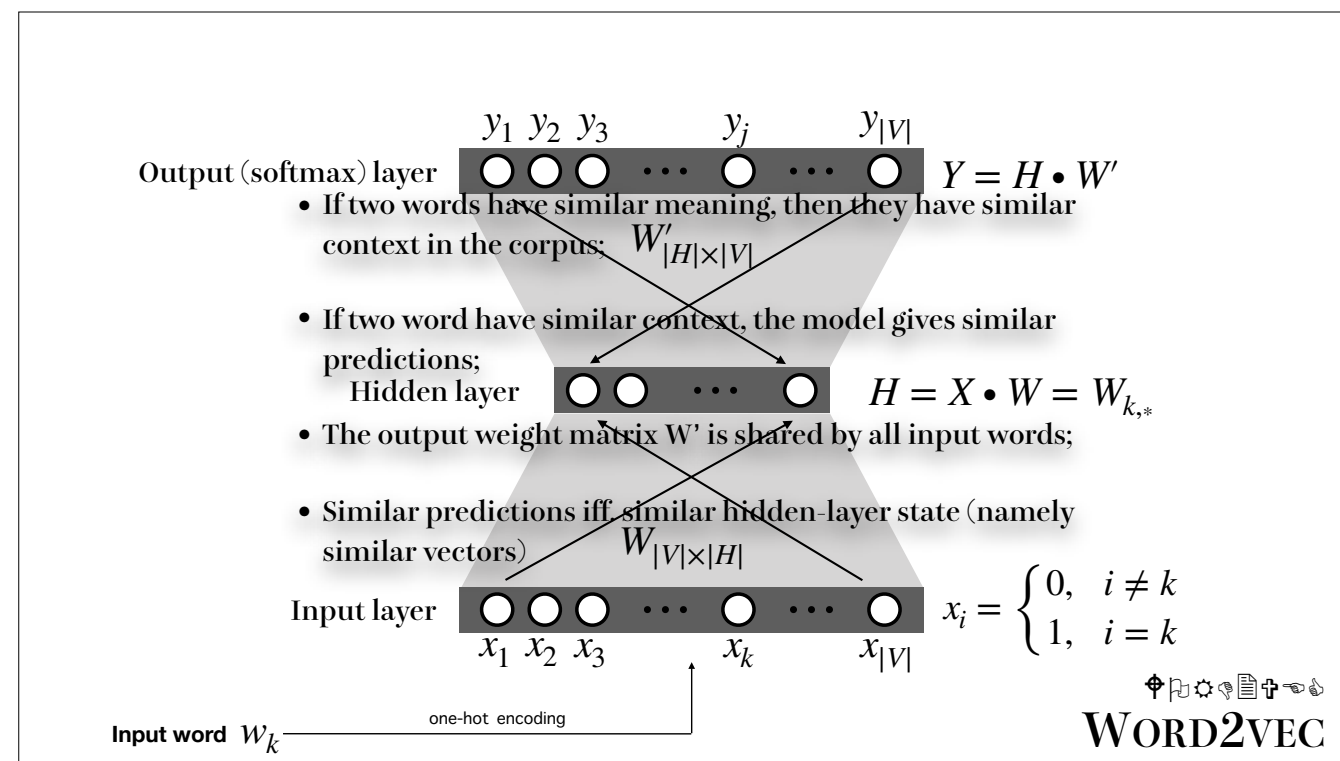
👉 ✂️ 🔄 👁️ ⚙️ 🖱️ 🎯 🗑️ 📁
EXPERIMENT 1

...

SO we cannot estimate CROSS-CORPUS semantic difference with the distance of embedded vectors from two SEPARATELY TRAINED word2vec models.
But we are actually very close to the solution. We just need to tweak the word2vec model a little bit to suit our task, which is the cross-corpus semantic comparison.



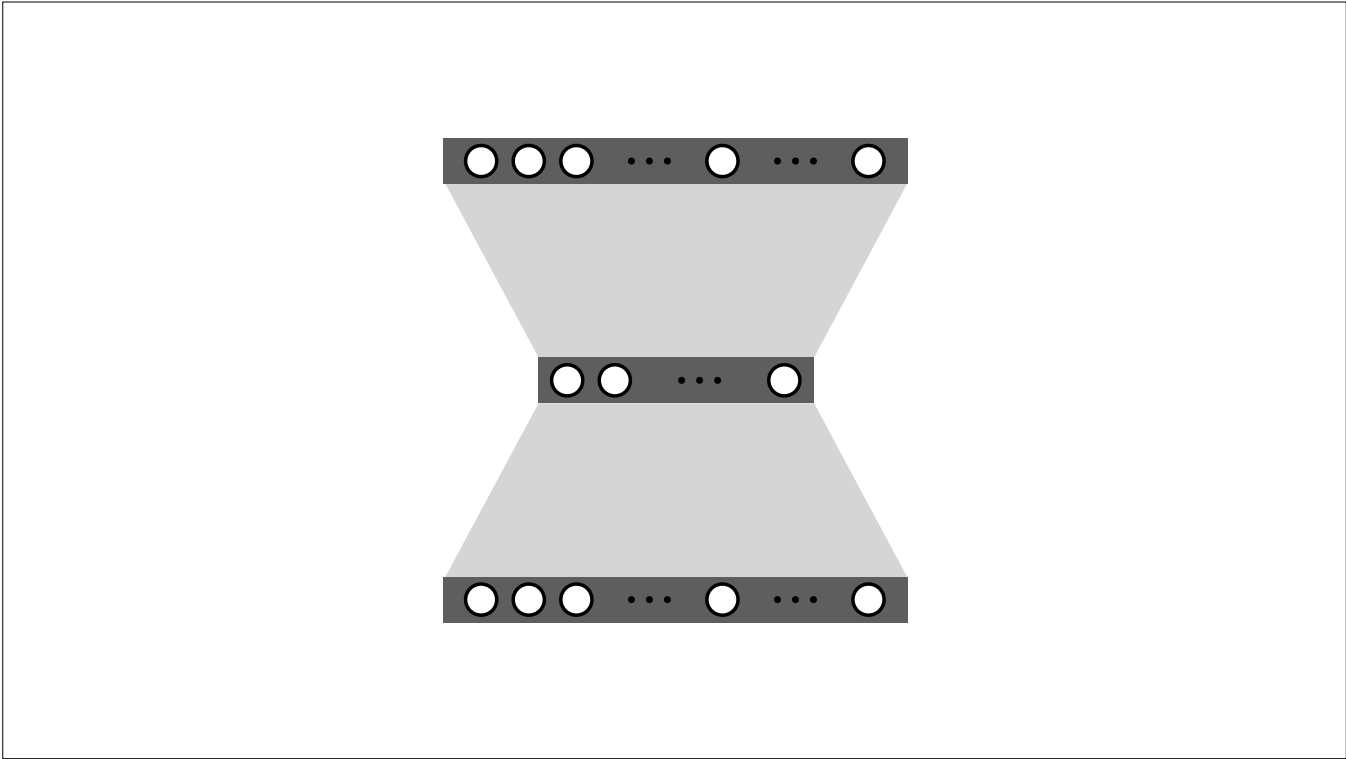
SO we cannot estimate CROSS-CORPUS semantic difference with the distance of embedded vectors from two SEPARATELY TRAINED word2vec models.
 But we are actually very close to the solution. We just need to tweak the word2vec model a little bit to suit our task, which is the cross-corpus semantic comparison.



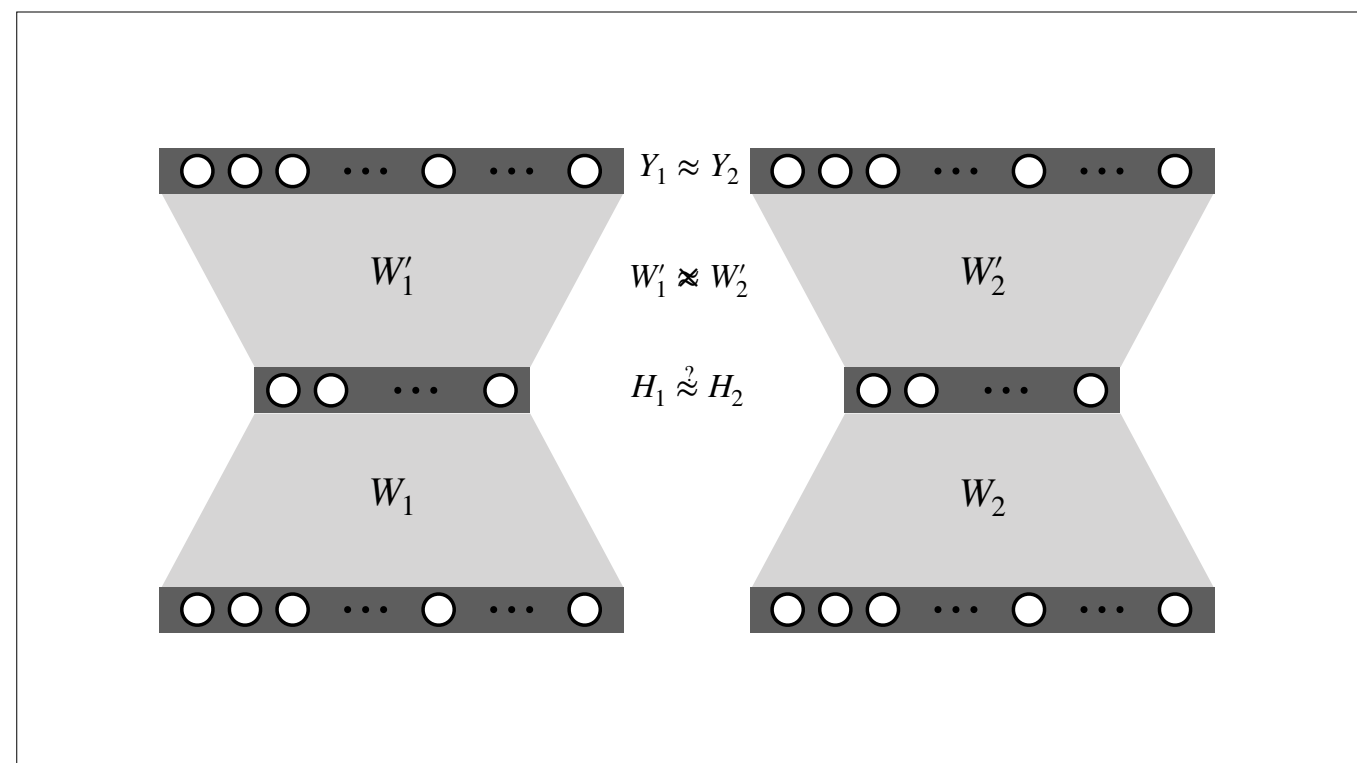
But we are actually very close to the solution. We just need to tweak the word2vec model a little bit to suit our task, which is the cross-corpus semantic comparison. To to this, we need to dig a little deeper into the Word2vec, and find out why word2vec doesn't work in this scenario

Let's look at its the prediction stage:

- A word in input with one-hot encoding
- It actually select a row of the input layer weight matrix, and feed to hidden layer H (embedded vector of the word)



Reason of Word2vec fails

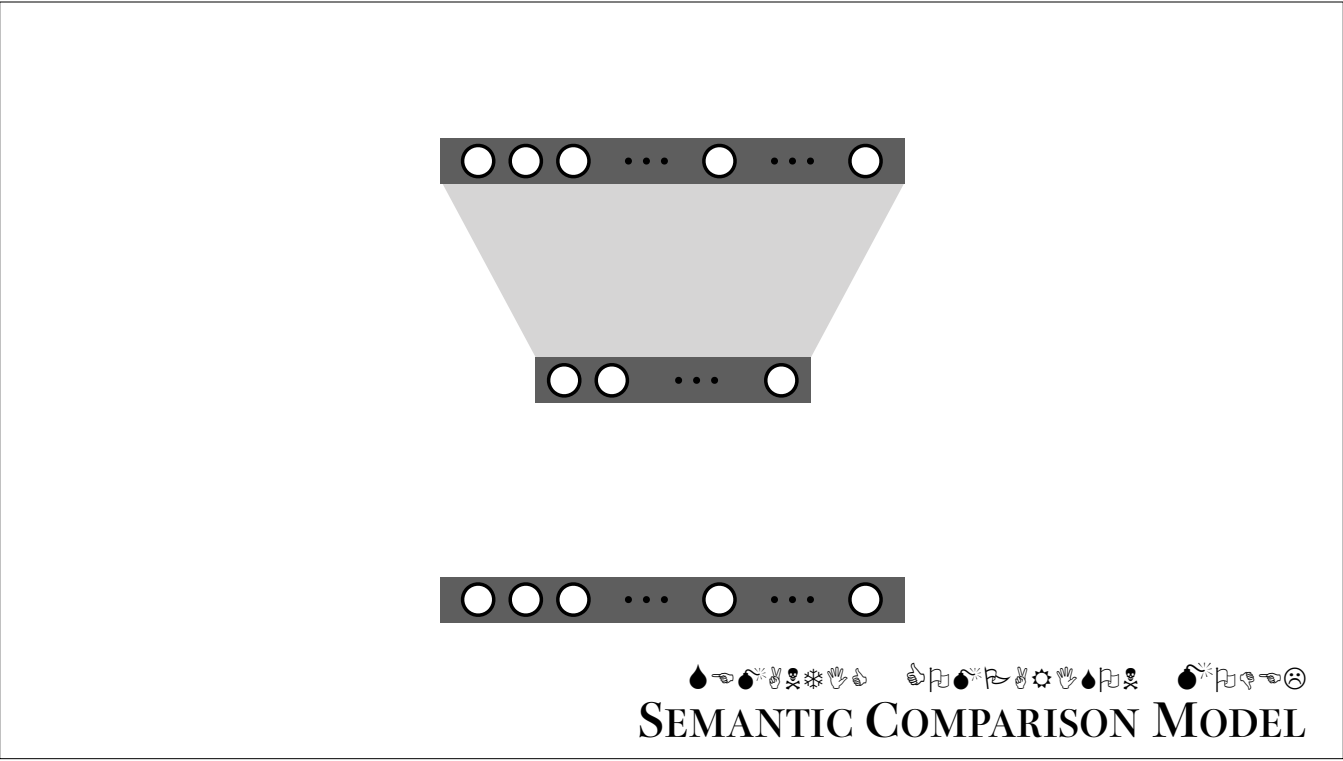


Reason of Word2vec fails

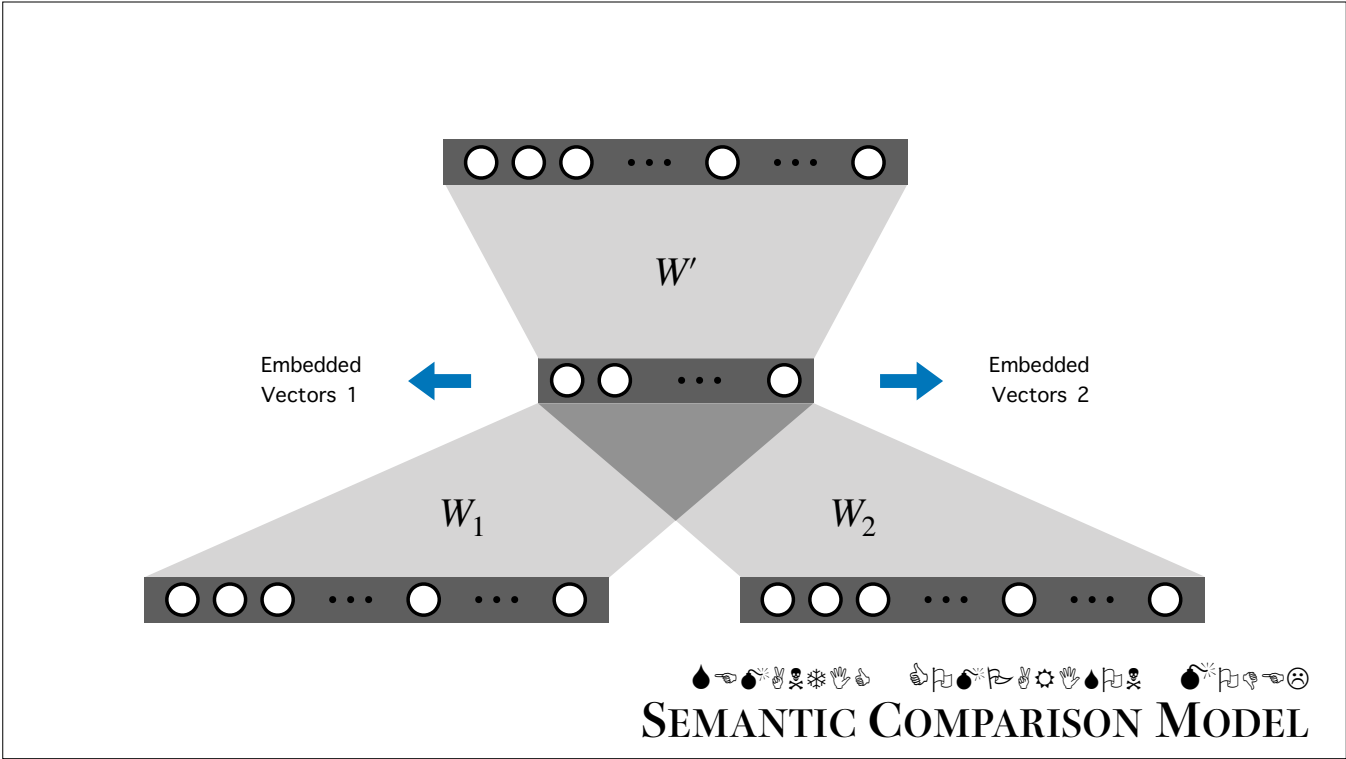
Semantic difference = distance of vectors
iff.
Vectors are associated with the same output-layer matrix W'

To understand the reason why word2vec doesn't work in this scenario, we need to dig a little deeper into the Word2vec. Let look at its the prediction stage:

- A



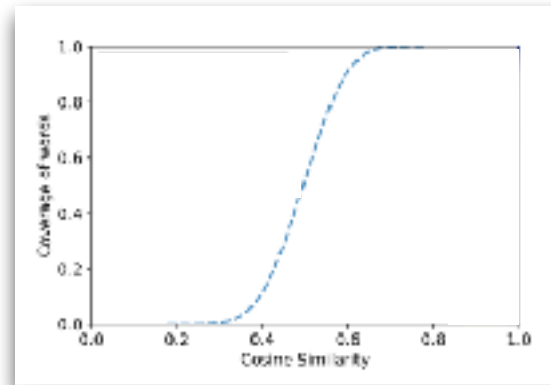
Reason of Word2vec fails



Word2vec to Semantic Comparison Model

Cross-corpus Semantic Comparison: Word2vec vs. SCM

(training set: Text8 & Text8)



$$\mu = 0.49/0.98, \sigma = 0.078/0.006$$



EXPERIMENT 1

we used Text8 as both input corpora for our SCM. For each word in the vocabulary, the model generated a pair of vectors, each representing its semantics in the corresponding corpus. Since the two input corpora here are identical, the cosine similarity of every vector pair should all be close to 1, if SCM can capture the words' semantics in both corpora correctly. Our experiment shows that for every word in the corpora, the average cosine similarity between its two vectors is 0.98, with a standard deviation 0.006.

As a reference, we trained a Word2Vec model on the same corpus twice, and calculated the cosine similarities between the vectors of the same words. Here the average similarity is 0.49 and standard deviation 0.078, indicating that the vectors from the two models cannot be compared, due to the training randomness

SCM Capturing Cross-corpus Semantic Difference

(Training set: Text8 & Text8_{syn})

- Synthesizing “jargons” using word replacement

replacing word pair	similarity
(chemist → archie)	0.65
(fl → proton)	0.56
(universe → wealth)	0.67
(educational → makeup)	0.66
(nm → furnicom)	0.45

$$\mu = 0.98, \sigma = 0.01$$



EXPERIMENT 2

cross-corpora semantic difference experiment

We randomly chose 5 words from the Text8 corpus and replaced them with 5 other words (see Table 2) to construct a new corpus Text8syn. In this way, these replacements become “jargons” of the original words in the new corpus Text8syn. Then we trained our architecture on Text8 and Text8syn,

all the replaced words were found to have small similarities in two corpora: the average similarity is 0.98 with a standard deviation of 0.01.

This experiment shows that our SCM is able to capture a word’s cross-corpora semantic difference.

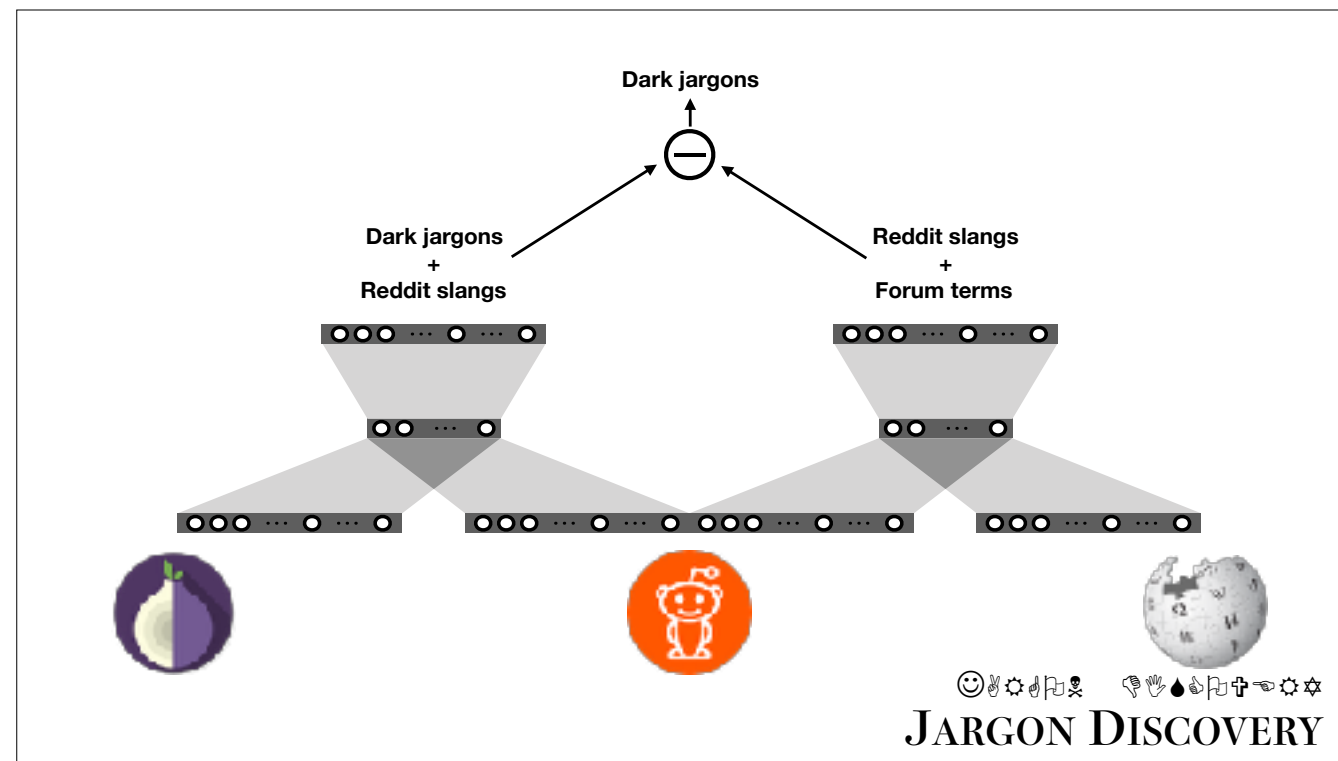
Vector quality: Word2vec vs. SCM

- Tomas Mikolov's evaluation code.
- Training set: Text8 vs. Text8+NULLED.
- Accuracy: 0.50 vs. 0.46.

👉 ⚡ 🔄 👉 ⚙️ 👉 🖐️ 👉 🌑 👉 🦴 ❄️ 📄
EXPERIMENT 3

In this experiment, we trained an SCM using Text8 along with a snapshot of Nulled [12], a collection of communication traces from an underground forum.

Tomas Mikolov [22] provides code and the test set for evaluating the quality of word vectors.



Reddit slangs: such as “damage” on reddit.com often appear during the discussion of VIDEO GAMES and as a result, its context becomes very much biased towards settings in the games (such as “heal”, “stun” and “dps”);

This the basic idea of jargon discover algorithm. It actually involves quite a few Implementation details and I don’t have time to cover all those in the talk, so plz refer to our paper for more details.

Key Idea: hypernymy

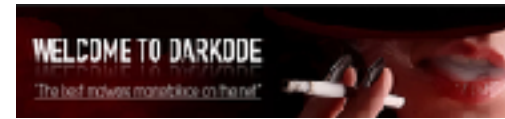
Different levels of hypernyms, e.g. cocaine -> stimulant -> drug

Different levels of hypernyms, e.g. cocaine -> stimulant -> drug

$$V_{\text{king}} - V_{\text{man}} = V_{\text{queen}} - V_{\text{woman}}$$

☺ 🐇 ⚙️ 📖 🦴 ⛪ 🦴 🐉 🗡️ ⚙️ 💧 ❄️ 🦴 🐉 🦴 🦴
JARGON UNDERSTANDING

Another interesting feature of word2vec is that: some kind of semantic relations can be calculated by arithmetics of embedded vectors



- 1.5 million communication traces, 117 million words
- 3,462 dark jargons covers 5 categories of illicit products
- Precision 0.91, recall 0.77

DATASETS & EVALUATION

Dataset: DARKNET MARKET ARCHIVES + “Identifying products in online cybercrime marketplaces: A dataset for fine-grained domain adaptation.”

- Silkroad: mostly drugs 6/2011 - 11/2013
- Darkode: cybercriminal wares, e.g. exploit kits, spam services, ransomware, and botnets. 3/2008 - 3/2013
- Hack Forum: cyber-security, hacking technology and others. 5/2008 - 3/2015
- NULLED: data stealing tools and services 11/2012 - 5/2016

We observe the 3,462 dark jargons covers 5 categories of illicit products: drugs has the most jargons.

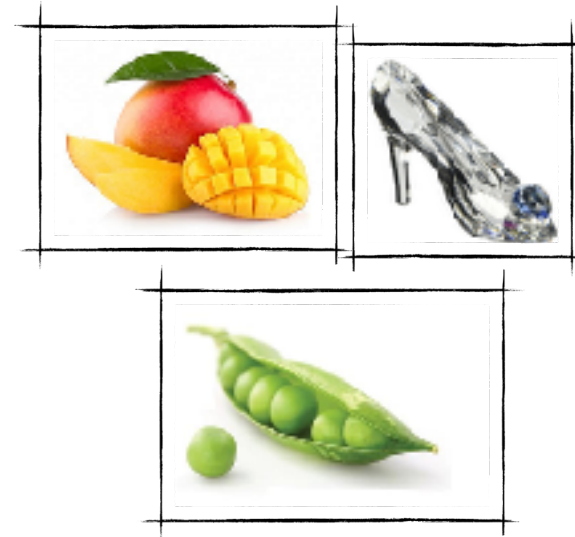
Evaluation

Precision: random sample 200 detected jargons

Ground truth set: Drug Enforcement Administration (DEA) drug codename list + 1,292 illegitimate products manually annotated


Recall: 774 jargon words in the set, 598 were successfully detected by Cantreader

FN: jargon “car” means “cocaine”, never used nowadays



DEA (Drug Enforcement Administration) drug code words (may 2017): we found many drug jargons are not included in the drug jargon lists recorded by DEA. For example, on average, around 25 dark jargons emerge each month on hack forums from 2010 to 2013.

- “cinderella” - a kind of cannabis
- “pea” - organic compound acts as a central nervous system stimulant
- “mango” - a kind of marijuana

canadabuds best uk weed vendor
5.0
•  Dave reviewed 1 year ago
• last edited 1 year ago
canadabuds best uk weed vendor nice cheese and hanig demcon haze also haze rare strains like cannabonic



We observe the 3,462 dark jargons covers 5 categories of illicit products: drugs has the most jargons.

Jargons can be used in the:

- profile of dealers and customers of illicit products,
- identify key players in the community and
- recover the ecosystem



Where are the dark jargons chosen from?

We observe cyber-criminals choose jargons from a variety of types of innocent-looking words (e.g. animal, plant, fictional character). 8 categories has over 30 jargons.

drug dealers like fruit (“pineapple”, “blueberry”, “lemon”)

hackers prefer mythological figures (“zeus”, “loki” , “athena”)

- We observe dark jargons also used in benign forums. (675 communication traces in Reddit related to illicit activity)
- We observe that dark jargons can help us find black words (dedicated used by cyber-criminals). We discover 522 black words with the help of discovered.

⌘⚙️🔍⚙️ ⌘🔍⚙️⚙️⚙️⚙️⌘⚙️
OTHER OBSERVATIONS

Measurement - the four forums
(8 types have more than 30 jargons).



Conclusion