machine learning

Text Analysis

Lecture IX

פיתוח:
משה פרידמן

Credits:
Yoav Goldberg, Ido Dagan, Reut Tsarfaty , Moshe Koppel, Wei Song, David Bamman, Ed Grefenstette, Chris Manning

A lot of Buzz words

- Text analysis
- Text processing
- Text mining
- Natural language processing (aka NLP)
- Computational linguistics

What do these terms mean?

How do they relate to machine learning?

What is a text?
A programmer's perspective

איך נתייחס לטקסט
בפרספקטיבה של
מתכנת?

What is a text?
A programmer's perspective

מחרוזת (וויקיפדיה) - בשפות תכנות מחרוזת היא טיפוס נתונים המכיל רצף של תווים.

ולכן בראיה של מתכנת – טקסט היא מחרוזת תווים.

What is a text?
A programmer's perspective

אך האם כל מחרוזת תווים משמעותה טקסט?

What is a text?
A programmer's perspective

אך האם כל מחרוזת תווים משמעותה טקסט?

```
l1vBCavOBjD22a5RFZQDn+jpr8D/zPS+hhXnXHazZAKkOx1qKbZtIZWDt3Mv
8TFt6YgNZ9RSN19r7G7Vim9q3l35EXbG1Vb/UwLY4WtcNLl9nHuvgclF8ebS
wn3gS+ArRdOa2KsSdg13LU1OsPsRmOlroPrS7jUM3jZS9g3FnukRRCYqJMZZ
nCf/2C330aeLHCLmNn6pegRqGb6hTy6717g6+xw/bGtLEIrd75AMcW+q72E/
x3UkMOx1wtRRougEfujI8YF/j2PM85YoZgW5HTc9+xrLxJ9Z8N13/TWujgC7
xmFO3VsP6IqfSZCvvatktwkXmvbf9S7e6uNzFs33Rw==
```

RSA encryption key

What is a text?
A programmer's perspective

אך האם כל מחרוזת תווים משמעותה טקסט?

What is a text?
A programmer's perspective

אך האם כל מחרוזת תווים משמעותה טקסט?

Accession number (Wikipedia)- In libraries, art galleries, museums and archives, an accession number is a unique identifier assigned to, and achieving initial control of, each acquisition

What is a text?

מסקנה – לא כל מחרוזת תווים היא טקסט.

אז מהו טקסט?

Text (Wikipedia) - In literary theory, a text is any object that can be "read“.

נחזור לטקסט עוד מעט

נבחן עוד כמה שאלות ...

מה הוא ידע?

Knowledge (Wikipedia) - Knowledge is a familiarity, awareness, or understanding of someone or something, such as facts (descriptive knowledge).

נבחן עוד כמה שאלות ...

מהי שפה?

Language is the use of a system of communication which consists of a set of sounds or written symbols.

<u>שפה טבעית</u> - להבדיל משפות אחרות (שפות תכנות למשל), שפה המבטאת תקשורת בין בני אדם נכנה שפה טבעית.

נבחן עוד כמה שאלות ...

מהי שפה?

Language is the use of a system of communication which consists of a set of sounds or written symbols.

הדרך בה אנחנו צורכים, מעבירים ומבינים ידע הוא דרך שפה

שפה טבעית ולמידת מכונה

Voice/Speech recognition /phonology

text

TAVILIETAKADUR

תביא לי את הכדור

שפה טבעית ולמידת מכונה

Voice/Speech recognition /phonology

OCR – Optical character recognition

שפה טבעית ולמידת מכונה

Voice/Speech recognition /phonology

OCR – Optical character recognition

Nonverbal / Gesture / Micro expression

credit Wei Song, et al

What do we mean by text?

Text (Wikipedia) - In literary theory, a text is any object that can be "read“.

String (Wikipedia) - In computer programming, a string is traditionally a sequence of characters.

Language is the use of a system of communication which consists of a set of sounds or written symbols.

A text string - We refer to text of (verbal) language saved as a string.

We do not include:

- Voice/Speech recognition/Phonology
- OCR – Optical character recognition
- Nonverbal Language / Gesture / Micro expression Recognition

What do we mean by text?

Text (Wikipedia) - In literary theory, a text is any object that can be "read".

String (Wikipedia) - In computer programming, a string is traditionally a sequence of characters.

Language is the use of a system of communication which consists of a set of sounds or written symbols.

A text string - We refer to text of (verbal) language saved as a string.

<u>We do include – (written) non-spoken language</u>:

- **Emoticons -** :-) :-(
- **Emoji -** ☺ 😢
- **Text messaging** – LOL ("laughing out loud"), "gr8" ("great")

And more

What do we know so far

Terms: Natural language and Text (verbal and non-verbal)

We want to process text, which describes natural language.

What's next?

- Bytes and bits
- Why is language processing hard?

Text as data

Some properties and challenges

- Variability - one meaning, many forms
- Ambiguity - one form, many meanings
- Zipf law

Levels of Language Processing

Text data – properties and challenges
Variability - one meaning, many forms

2maro 2marrow 2mor 2mora

2moro 2morow 2morr 2morro 2morrow 2mr

2mro 2mrrw 2mrw 2mw tmmrw tmo tmoro tmorrow

tmoz tmr tmro tmrow tmrrow tmrrw tmrw tmrww tmw

tomaro tomarow tomarro tomarrow tomm

tommarow tommarrow tommoro tommorow

tommorrow tommorw tommrow tomo tomolo tomoro

tomorow tomorro tomorrw tomrw

Text data – properties and challenges
Variability - one meaning, many forms

he acquired it

he purchased it

he bought it

it was bought by him

it was sold to him

she sold it to him

she sold him that

Text data – properties and challenges
Ambiguity - one form, many meanings

Bank

noun

noun

Text data – properties and challenges
Ambiguity - one form, many meanings

Time Flies

If time is a **noun**

If time is a **verb**

Text data – properties and challenges
Ambiguity - one form, many meanings

Ambiguity – (extreme cases)

* אם מישהו רכב על סוס ועזב, אז הוא פרש או לא?
* להודות למישהו בפה מלא, זה מנומס או לא?
* אם ארכיאולוג העלה חרס בידו, הוא הצליח או לא?
* באיסטנבול לא סוגרים דלתות. הם טורקים.
* שקלתי הרבה לפני שהחלטתי לעשות דיאטה

Text data – properties and challenges
Zipf law

Word frequencies follow a power-law distribution.
--> Long tail - most words will occur only few times if they occur

There are very likely to be word forms we did not see

In a 43M words text, there are:
- 316K unique words
- 144K words occur once
- 42K words occur twice
...
- 26K words occur >50 times

Levels of Language Processing

Phonology

TAVILITAKADUR → תביא לי את הכדור

Levels of Language Processing

Phonology

Morphology

Independently

Levels of Language Processing

Phonology

Morphology

בגרמנית – אפשר להרכיב מילים מורכבות מאוד בעלות תחיליות, אמצעיות וסופיות.

lebensversicherungsgesellschaftsangestellter

Life insurance company employee

Levels of Language Processing

Phonology

Morphology

Syntax

The dog bit the man

Levels of Language Processing

Phonology

Morphology

Syntax

Semantics

The dog bit the man

exists x,y:
    Dog(x)
    Man(y)
BITE(x,y)

Machine Learning and Text Analysis

Machine Learning problems

Binary	$y \in \{-1, 1\}$	classification problems
Multi-class	$y \in \{1, 2, \ldots, k\}$	
Multi-label	$y \in 2^{\{1, 2, \ldots, k\}}$	

Also

Regression

And others …

Machine Learning problems – for different language levels

The basic units of text processing:

- Corpus - document collection
- Document
- Section
- Paragraph
- Sentence
- Phrase
- Word

Machine Learning problems – for different language levels

Document --> label

- Language classification
- Topic classification
- Author classification
- Sentiment classification

Sentence --> label

- Usually the same as in "document --> label"
  shorter text --> harder task

Question: are each of these binary / multi-class / multi-label?

Machine Learning problems – for different language levels

Word --> label

- Tokens vs. Types

<u>Type --> label</u>

  - Sad vs happy words
  - Adjectives vs. Nouns

<u>Token --> label</u>

  - Sentence boundary detection
  - Common spelling mistakes ("then" vs "than")

Text Analysis and Machine Learning

Text analysis generic flow:

Acquire an annotated corpus (will become a dataset)

- Split to train and test/validation
- Vectorization (will be explain soon)
- Train a machine learning model on train set
- Tune model/ Check quality

Text as features

How do we extract features from text?

What is the basic atom level of text to extract as features?

How do we take into account any text properties?

Basic Text Analysis Flow

Sentence Boundary Detection

Sentence Boundary Detection

Perhaps the most basic task.

- Non-trivial.
- Need to consider corpus, features, annotation procedure, biases.
- Need to think of your use-cases, choose your sentence definition, methods, and trade-offs

What is a sentence?

In a quiet voice, he said

"this will not work. I am quitting",

and the he left the room.

Sentence Boundary Detection

sentences = text.split(". ")?

Sentence boundary detection: placing sentence boundaries on a text.

What is a sentence?

- “Do you want to go?” said Jane.
- Mr. Collins said he was going.
- He lives in the U.S. John, however, lives in Canada.

What is a word?

sequence of characters

basic unit of meaning?

white-space bounded?

→ whitespace might not be enough

Using tokens instead of words

Types and tokens

Type = abstract descriptive concept Token = instantiation of a type

- a token could contain a couple of words

To be or not to be

- 6 tokens (to, be, or, not, to, be)
- 4 types (to, be, or, not)

Types = the vocabulary; the unique tokens

Tokenization

Given a character sequence and a defined document unit, tokenization is the task of chopping it up into pieces, called *tokens* (perhaps at the same time throwing away certain characters), such as punctuation.

Here is an example of tokenization:

<u>Input</u>:

Friends, Romans, Countrymen, lend me your ears;

<u>Output</u>:

- Friends
- Romans
- Countrymen
- lend
- me
- your
- ears

Tokenization: dividing a text into tokens.

Words, Sentences and Punctuation

We typically don't want to just strip all punctuation, however.

- Punctuation signals boundaries (sentence, clausal boundaries, parentheticals, asides)
  - Some punctuation has illocutionary force, like exclamation points (!) and question marks (?)
  - Emoticons are strong signals of e.g., sentiment
- Most tokenization algorithms (for languages typically delimited by whitespace) use regular expressions to segment a string into discrete tokens.

What is a word?
Twitter Text example

sequence of characters

basic unit of meaning?

white-space tokenized?

@dbamman have you seen this :) http://popvssoda.com

- @
- dbamman
- have
- you
- seen
- this
- :
-)
- http
- :
- /
- /
- popvssoda
- .
- com

Difficulties

- When does punctuation disrupt the desired boundaries of a token?

Emoticons	:) :D \o/ o_O
URLs	http://www.google.com
Prices	$19.99
Decimals	19.99
Hyphens	state-of-the-art
Usernames	@dbamman
Hashtags	#blacklivesmatter

Basic Text Analysis Flow

Word / token normalization

Punctuation

- U.S.A. → USA (acronym)
- Ltd. → Ltd

  Risk:

דר <-- ד"ר

Word / token normalization

Case folding

- General Motors → general motors

Risk:

US → us

Basic Text Analysis Flow

Word / token normalization

Morphology

Morphemes -
The small meaningful units that make up words

- Lemma: The core meaning--bearing units
- Affixes: Bits and pieces that adhere to stems
  - Often with grammatical functions

Morphological analysis

Lemma: the "dictionary entry" of a word.

create, created, creating, creator, creativity

↓ ↓ ↓ ↓ ↓

create, create, create, creator, creativity

ספרו - ה-ספר-של-הוא

Lemmatization: reducing the inflected forms of a word into a single form for easy analysis.

Stem: a "base form", based on heuristics.

create, created, creating, creator, creativity

↓ ↓ ↓ ↓ ↓

creat, creat, creat, creat, creat

Stemming: cutting the inflected words to their root form.

Function /stop words removal

Stop words – usually refers to the most common words in a language

Function words – usually refers to words that have little meaning, such as articles (e.g., *a, the*) and conjunctions (e.g., *and, but*)

Stop word / Function word removal: removing the words, which are considered as stop-words or function-words.

A	It	These
About	Its	They
Again	Itself	This
All	Just	Those
Almost	km	Thus
Also	Made	To
Although	Mainly	Upon
Always	Make	Use
An	May	Used
And	mg	Using
Another	Might	Various
Any	ml	Very
Are	mm	Was
As	Most	We
At	Mostly	Were

Basic useful preprocessing operation - summary

Lemmatization: reducing the inflected forms of a word into a single form for easy analysis.

Stemming: cutting the inflected words to their root form.

Tokenization: dividing a text into tokens.

Sentence breaking: placing sentence boundaries on a text.

Word Normalizations and cleaning

Stop word removal: removing words, which are considered as such.

<u>Other Useful operations:</u>

Part-of-speech tagging: identifying the part of speech for every word.

Text as data – properties and challenges

Sparse data

- Zipf's law, variability

Symbolic

- abstract symbol to meaning mapping, variability, ambiguity

Many levels of granularity

- document, paragraph, sentence, word, characters

How would these affect a classifier over text data?

Basic Text Analysis Flow

Vectorization: from document to feature vector

The bag of words (BOW) model:

Each **word** is treated as a feature in a unit called **document**.

Each such word will become a feature

How do we measure their strength?

- Word Count
  - What about zipf law?

Vectorization: extracting basic feature units

The bag of words (BOW) model:

Each **word** is treated as a feature in a unit called **document**.

Each such word will become a feature

- Alternative: **original tokens – no processing**
- Alternative: **normalized words – e.g., lemmas, stems**
- Alternative: **characters – e.g., prefixes**
- Alternative: **ngrams – unigram, bigram, trigram**
- More complex alternatives …

Vectorization: extracting basic feature units

The bag of words (BOW) model:

Each **word** is treated as a feature in a unit called **document**.

Each such word will become a feature

- Alternative: **normalized words – e.g., lemmas, stems**

Vectorization: extracting basic feature units

The bag of words (BOW) model:

Each **word** is treated as a feature in a unit called **document**.

Each such word will become a feature

- Alternative: **ngrams – unigram, bigram, trigram**

ngrams: unigrams bigrams trigrams

['the','special', 'onion', 'soup', 'was', 'not', 'very', 'bad',
'the special', 'special onion', 'onion soup', 'soup was',
'was not', 'not very', 'very bad', 'the special onion', 'special onion soup',
'onion soup was', 'soup was not', 'was not very', 'not very bad']

Basic Text Analysis Flow

Vectorization: feature's value

Count Vectorizer - Converts a collection of text documents to a matrix of token counts

The bag of words (BOW) model:

Word Count

- Binary – word occurs in document? $\in \{0, 1\}$
- Raw count = $f_{t,d}$ - number of occurrences in document
- term frequency = $\mathrm{tf}(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}},$
- Log scaled – zipf law = $\log(1 + f_{t,d})$

Vectorization: feature's value

Tf·idf Vectorizer - Converts a collection of raw documents to a matrix of TF-IDF features

The bag of words (BOW) model:

Word Count

- Binary, $f_{t,d}$, $\mathrm{tf}(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}$, $\log(1 + f_{t,d})$

How rare is the word?

inverse document frequency (idf) = $\log \frac{N}{|\{d \in D : t \in d\}|}$

number of documents

number of documents containing term

$$\mathrm{tfidf}(t,d,D) = \mathrm{tf}(t,d) \cdot \mathrm{idf}(t,D)$$

Basic Text Analysis Flow

Vectorization: Dictionary based features

- If we have word-lists, might as well use them.
- "how many times a ``negative word`` appears?"
- "how many times a name of a city in Israel occurs"?

The dictionary-based features, are different basic unit features (unlike regular tokens/n-grams).
* Detected differently
* Processed differently
* Could also be manually vectorized differently

Vectorization: Dictionary based features

- If we have word-lists, might as well use them.
- "how many times a ``negative word`` appears?"
- "how many times a name of a city in Israel occurs"?

Sentiment Dictionaries

pos	neg
unlimited	lag
prudent	contortions
supurb	fright
closeness	lonely
impeccably	tenuously
fast-paced	plebeian
treat	mortification
destined	outrage
blessing	allegations
steadfastly	disoriented

Vectorization: Dictionary based features

Affective state dictionary

State	Definitions	Terms
Emotion	Relatively brief episode of response to the evaluation of an external or internal event as being of major significance.	angry, sad, joyful, fearful, ashamed, proud, elated, desperate
Mood	Diffuse affect state, most pronounced as change in subjective feeling, of low intensity but relatively long duration, often without apparent cause.	cheerful, gloomy, irritable, listless, depressed, buoyant
Interpersonal stance	Affective stance taken toward another person in a specific interaction, coloring the interpersonal exchange in that situation.	distant, cold, warm, supportive, contemptuous, friendly
Attitude	Relatively enduring, affectively colored beliefs, preferences, and predispositions towards objects or persons.	liking, loving, hating, valuing, desiring
Personality traits	Emotionally laden, stable personality dispositions and behavior tendencies, typical for a person.	nervous, anxious, reckless, morose, hostile, jealous

DictVectorizer - Transforms lists of feature-value mappings to vectors.
* This type of vectorizer might be used for dictionary-based features
* Significance and normalization could also be applied (such as tf-idf)

Basic Text Analysis Flow

Advanced features: word vectors

Word Vectors

... and the *cute* **kitten** *purred* and then ...
... the *cute furry* **cat** *purred* and *miaowed* ...
... that the *small* **kitten** *miaowed* and she ...
... the *loud furry* **dog** *ran* and *bit* ...

Example **basis vocabulary**: *{bit, cute, furry, loud, miaowed, purred, ran, small}*.
kitten context words: {cute, purred, small, miaowed}.
cat context words: {cute, furry, miaowed}.
dog context words: {loud, furry, ran, bit}

$$\mathbf{kitten} = [0, 1, 0, 0, 1, 1, 0, 1]^\top$$
$$\mathbf{cat} = [0, 1, 1, 0, 1, 0, 0, 0]^\top$$
$$\mathbf{dog} = [1, 0, 1, 1, 0, 0, 1, 0]^\top$$

Advanced features: word vectors

Word Vectors

... and the *cute* **kitten** *purred* and then ...
... the *cute furry* **cat** *purred* and *miaowed* ...
... that the *small* **kitten** *miaowed* and she ...
... the *loud furry* **dog** *ran* and *bit* ...

Example **basis vocabulary**: *{bit, cute, furry, loud, miaowed, purred, ran, small}*.
kitten context words: {cute, purred, small, miaowed}.
cat context words: {cute, furry, miaowed}.
dog context words: {loud, furry, ran, bit}

Cooccurrences :

Measure cooccurrences of words with other words
E.g., PPMI – positive pointwise mutual information

$$\mathrm{PPMI}(w,c) = \max(\log_2 \frac{P(w,c)}{P(w)P(c)}, 0)$$

$$\mathbf{kitten} = [0,1,0,0,1,1,0,1]^\top$$
$$\mathbf{cat} = [0,1,1,0,1,0,0,0]^\top$$
$$\mathbf{dog} = [1,0,1,1,0,0,1,0]^\top$$

Advanced features: word vectors - LSI

LSI – SVD like PCA, but done on **term-document matrix** (rather than the covariance matrix)

In LSI we decompose the **term-document matrix** *A* into a product of three matrices: **The term matrix, The document matrix, The singular value matrix**

C	d_1	d_2	d_3	d_4	d_5	d_6
ship	1	0	1	0	0	0
boat	0	1	0	0	0	0
ocean	1	1	0	0	0	0
wood	1	0	0	1	1	0
tree	0	0	0	1	0	1

$=$

U	1	2	3	4	5
ship	−0.44	−0.30	0.57	0.58	0.25
boat	−0.13	−0.33	−0.59	0.00	0.73
ocean	−0.48	−0.51	−0.37	0.00	−0.61
wood	−0.70	0.35	0.15	−0.58	0.16
tree	−0.26	0.65	−0.41	0.58	−0.09

$\times$

Σ	1	2	3	4	5
1	2.16	0.00	0.00	0.00	0.00
2	0.00	1.59	0.00	0.00	0.00
3	0.00	0.00	1.28	0.00	0.00
4	0.00	0.00	0.00	1.00	0.00
5	0.00	0.00	0.00	0.00	0.39

$\times$

V^T	d_1	d_2	d_3	d_4	d_5	d_6
1	−0.75	−0.28	−0.20	−0.45	−0.33	−0.12
2	−0.29	−0.53	−0.19	0.63	0.22	0.41
3	0.28	−0.75	0.45	−0.20	0.12	−0.33
4	0.00	0.00	0.58	0.00	−0.58	0.58
5	−0.53	0.29	0.63	0.19	0.41	−0.22

term-document matrix

term matrix - one (row) vector for each term

singular value matrix – diagonal matrix with singular values, reflecting importance of each dimension

document matrix - one (column) vector for each document

Text Analysis and Machine Learning

Text analysis generic flow:

Acquire an annotated corpus (will become a dataset)

- Split to train and test/validation
- Vectorization – representing and counting words
- Train a machine learning model on train set
- Tune model/ Check quality

Text analysis example: spam detection

Text analysis example: spam detection

Spam detection is basically a text cat problem, but there are some special issues:

- Collecting data – non-spam email is private
- Asymmetry – must never class good mail as spam
- Adversarial – spammers try to defeat filters

Text analysis example: spam detection

Data acquisition:

- gmail has user feedback
  - LOTS of examples
  - Haphazardly labeled
  - How much info do they keep about each email?

Text analysis example: spam detection

Problem of False Positives:

- False positives more costly than false negatives
- Research must report recall-precision curves; key point is precision ~ 1

Adversarial Problem

- Spammers reverse engineer global filters; use nasty tricks to circumvent them
- This is what makes spam detection an interesting problem

Basic Spam

It cost you nothing (Yes! $0) to give Us a call, We will contact You back

Absolutely No exams/Tests/classes/books/Interviews
No Pre-School qualification Needed!

Inside USA: 1-718-989-5XXX
0utside USA: +1-718-989-5XXX

Degree, Bacheelor, masteerMBA, PhDD available in the field of your choice that's Right, You can even become a doctor & receive all the benefits That omes With it!

Please Leave Below 3 INFO in voicemail:

1) your Name
2) your Country
3) your Phone No. (with Countrycode)

Call Now! 24 hours a day, 7 Days a week to recieve Your call

Basic Spam

Most Honorable Sir,

I am Ehud Olmert, formerly the Prime Minister of Israel. I URGENTLY REQUIRE YOUR ASSISTANCE IN A MOST DISCRETE MATTER. As a result of certain events in my country, it has become necessary for me to transfer a considerable sum of cash to a foreign bank account. I turn to you as a MOST HONORABLE AND TRUSTED PERSON for your discrete assistance.

The total amount involved is THIRTY MILLION NEW ISRAELI SHEKELS only [30,000.000.00 NIS] and we wish to transfer this money into safe foreigners account abroad. I am only contacting you as a foreigner because this money cannot be approved to a local person here, but to a foreigner who has information about the account, which I shall give to you upon your positive response. I am revealing this to you with believe in God that you will never let me down in this business, you are the FIRST AND THE ONLY PERSON that I am contacting for this business, so please reply urgently so that I will inform you the next step to take urgently.

At the conclusion of this business, you will be given 40% of the total amount, 50% will be for us while 10% will be for the expenses both parties may incurred during this transaction. PLEASE, TREAT THIS PROPOSAL AS TOP SECRET.

Early Work Sahami et al '98

- Learner: Naïve Bayes
- Feature Set: Words, Phrases, Structural Features
- Feature Selection: top 500 infogain
- Evaluation Data: ~1700 Messages, ~88% Spam
- Results: Spam precision 100%, Spam recall 98.3%

Early Work Sahami et al '98

Hand Crafted Features

- 35 Phrases
  - 'Free Money'
  - 'Only $'
  - 'be over 21'
- 20 Domain Specific Features
  - Domain type of sender (.edu, .com, etc)
  - Sender name resolutions (internal mail)
  - Has attachments
  - Time received
  - Percent of non-alphanumeric characters in subject

Later Studies

- The early work was followed by the usual stream of extended feature sets and fancier learning methods (e.g. SVM)
- It is now common to use over 100,000 features
- Learning methods for huge data sets must be very efficient (online algorithms)
- Methods must be adaptive

Some Spammers' Tricks

- Fill messages with real text taken from books, sites, etc.
- Encoded Text
- Distorted Text

Secret Decoder Ring Dude

- Another spam that looks easy

Online Pharmacy - 24/7 Customer Care

Hundreds of products for dozens of ailments. We carry everything from Pain Relief to Skin Care products. Our most popular include:

- Viagra - Proven sexual aid to enhance performance
- Soma - The best in muscle relaxation available
- Phentermine - Safe, proven way to reduce weight
- ... and more!

Visit the Online Pharmacy for your medical needs

Please unsubscribe me

- Is it?

Secret Decoder Ring Dude

Pharmacy

Produc<!LZJ>t<!LG>s

- HTML word brea


```
<TABLE WIDTH=500 CELLPADDING=5 CELLSPACING=0><TR><TD BGCOLOR="PINK"
 </TD></TR><TR><TD BGCOLOR="PINK"> </TD><TD> <CENTER><FONT SIZE=5
Phar&#109;acy</B></FONT><FONT SIZE=4 FACE="ARIAL"><B> - 24/7
C&#117;<!LG>s<!LZJ>to&#109;er Care</B></FONT></CENTER><BR> <FONT SIZE=3 FACE="ARIAL"><B>
H&#117;ndred<!LG>s of prod&#117;c<!LZJ>t<!LG>s for dozen<!LG>s of ail&#109;en<!LZJ>t<!LG>s. We carry
every<!LZJ>thin&#103; fro&#109; Pain Relief <!LZJ>to Skin Care prod&#117;c<!LZJ>t<!LG>s. O&#117;r
&#109;o<!LG>s<!LZJ>t pop&#117;lar incl&#117;de:<BR><BR> <UL> <LI>Via&#103;ra - Proven
<!LG>sex&#117;al aid <!LZJ>to enhance perfor&#109;ance! <LI>So&#109;a - The <!AJ>&#98;e<!LG>s<!LZJ>t in
&#109;&#117;<!LG>scle relaxa<!LZJ>tion availa<!AJ>&#98;le! <LI>Phen<!LZJ>ter&#109;ine - Safe, proven way
<!LZJ>to red&#117;ce wei&#103;h<!LZJ>t! <LI>... and &#109;ore! </UL><BR> <CENTER><A
HREF="HTTP://www.oneni&#103;htpill&#98;iz/c6611"><FONT SIZE=4 FACE="ARIAL"><B><U>Visit the
Online Phar&#109;acy for yo&#117;r &#109;edical needs</U></B></FONT></A></CENTER><BR>
</B></FONT></TD><TD BGCOLOR="PINK"> </TD></TR><TR><TD BGCOLOR="PINK"
COLSPAN=3> </TD></TR></TABLE><BR> <FONT SIZE=1 COLOR="WHITE">ctkwr zirwt
fyzv&#103;<BR>&#117;vf&#117;&#109; npeer zkntc<BR>depeo fvtnt</FONT><BR><A
HREF="HTTP://www.7x24clickhere.co&#109;/re&#109;ove/"><FONT SIZE=1 COLOR="BLACK"><U>Please
&#117;ns&#117;&#98;scri&#98;e &#109;e</U></FONT></A><BR><FONT SIZE=1 COLOR="WHITE">toxy&#103;
iwxwy vhthf<BR>&#117;i&#106;&#109;a &#98;&#109;xih i&#109;wd&#117;<BR>r&#109;whd
fyr&#109;&#103;</FONT>
```

Diploma Guy

Dplmoia Pragorm

Caerte a mroe prosoeprus

Dlpmoia Pragorm

Caerte a mroe prosoeprus frtuue for yorsuelf

Recieve a flul doplima from non areceditcd
unvtersiiies baesd upon your rael lfie exneriepce

You will not be tseted, or irteeviewnd
Receive a Msetar's, Bachelor's or Dtctarooe

Clal 24 horus a day 7 days a week

1 - 2 7 0 - 8 1 7 - 8 2 4 7

More of Diploma Guy

```
lzqqbdhp
'U'''D'''M'''1
'n''''i''''a''''
'i'''p'''s'''2
'v'''l'''t'''7
'e'''o'''e'''0
'r'''m'''r''''
's'''a'''s'''8
'i'''s''''''''1
't''''''''p'''7
'y''''''''H''''
''''''''''D'''8
''''''''''''''2
''''''''''B'''4
''''''''''a'''7
vvsmll
```

One Pretty Good Text Cat Method

- Optimally compress spam training examples
- Optimally compress non-spam training examples
- Check which compression method better compresses suspicious message

Why This Works

- Works at level of character n-grams
- Should be applied to html source
- Captures weird encodings, word distortions
- Probably using character n-grams with SVM would also work well