Adventures in Unsupervised Learning:

Deciphering the Voynich manuscript

Based on work by Bradley Hauer and Grzegorz Kondrak. TACL'16

Dan Goldwasser

Purdue University

dgoldwas@purdue.edu

Introduction

- *Voynich manuscript is a medieval codex*
  - *The manuscript was dated to 1404-1438AD*
- Some characteristics
- Open problem for centuries
  - Some failed attempts
- Recent breakthrough by two NLP researchers
  - NLP?

Introduction

- **Natural Language Processing?**
  - *The way computers can deal with human language*
  - Quick review
- **Code breaking**
  - Very similar to natural language processing
  - Quick review
- ***Deciphering the Voynich Manuscript using NLP methods***
  - Methods and initial results.

Natural Language Processing

Purdue *in the News*

Google purdue news

Web News Shopping Images Videos More Search tools

About 73,100 results

About 73,100 results (0.31 seconds)

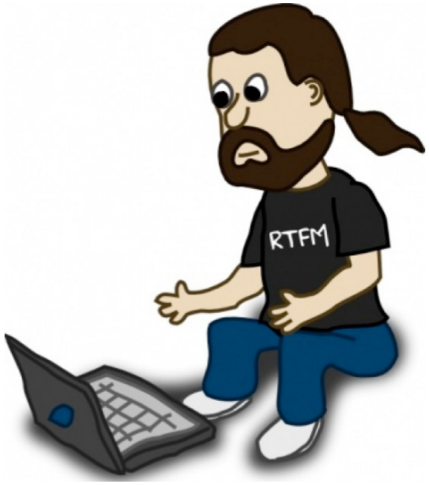
Purdue's Discovery Park launches global soundscapes re…
Purdue Newsroom - Oct 29, 2014
WEST LAFAYETTE, Ind. - **Purdue** University ecologist Bryan Pijanowski gained international attention for an Earth Day effort to capture ...

Purdue volleyball loses to Illinois
wlfi.com - Oct 25, 2014
CHAMPAIGN, Ill. (**Purdue** Sports) — The No. 13 Boilermaker volleyball team battled No. 10 Illinois to the wire in three sets of a four-set loss on ...

Illinois Volleyball block beats Indiana, No. 13 **Purdue**
Daily Illini - Oct 26, 2014

NLP in Practice

- **Sentiment Analysis**
  - Meaningful interpretation of product reviews
  - Identify the product aspects users care about
  - *Deception detection*



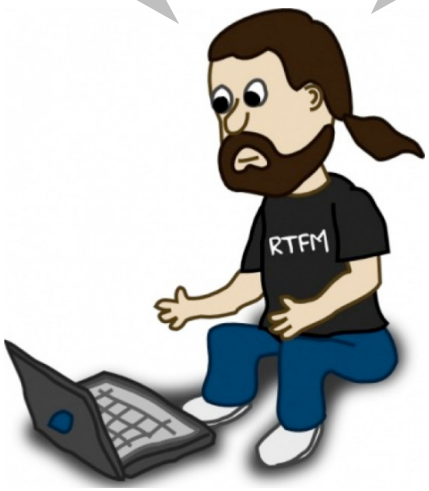
I just bought *company-A* newest laptop. The display is *awesome*, the speakers are *not that great*. I'm *happy* with the performance, but I think they charge too much for it!

Display: Positive
Speakers: Negative
Performance: Positive
Price: Negative

NLP can be Very Challenging!

Dude, I just watched this horror flick! Selling points: nightmares scenes, torture scenes, terrible monsters that was so bad a##!

Don't buy the popcorn it was terrible, the monsters selling it must have wanted to torture me, it was so bad it gave me nightmares!



More than Words

Natural language is inherently ambiguous

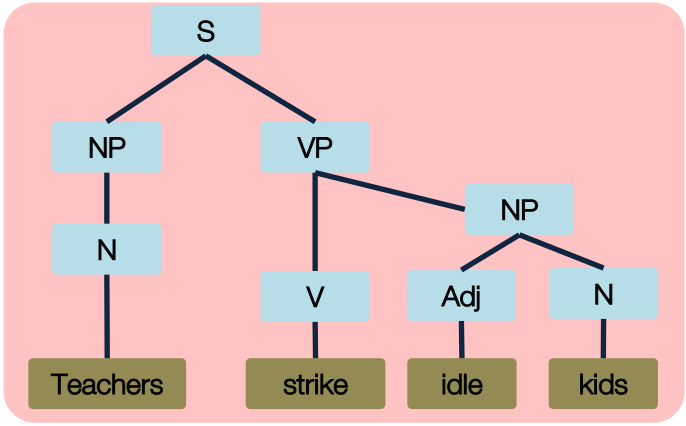
"Teachers strike idle kids"

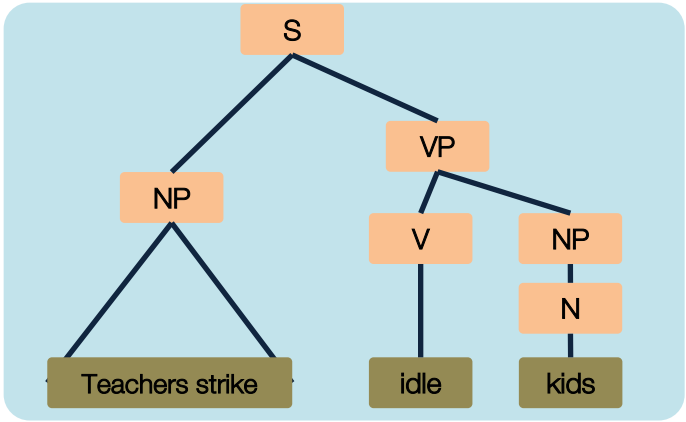
I just watched this horror flick!
.. nightmares ..torture ..,
terrible ..monsters

Horror movie + nightmare = Good

don't buy the popcorn it was
terrible.. monsters …torture
..nightmares!

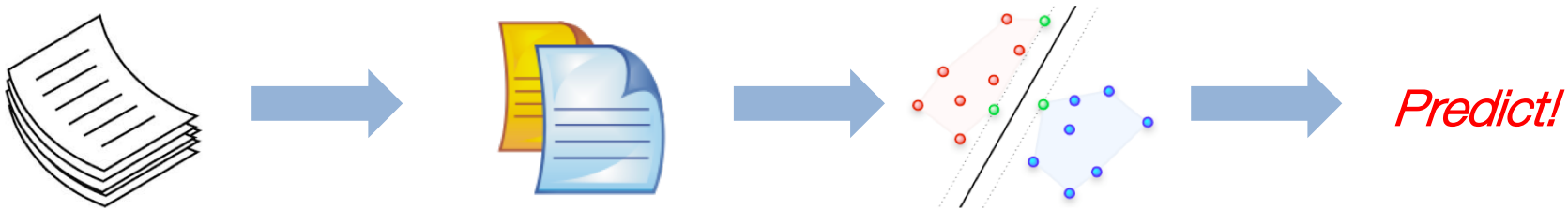
Food + nightmare = Bad



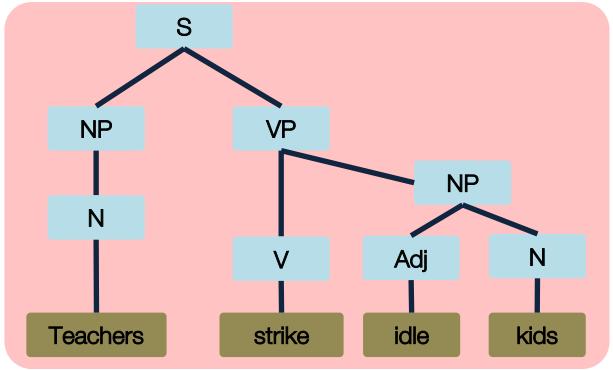


Dealing with Ambiguity

- ***Machine learning*** is an effective tool for resolving ambiguities
  - Build statistical prediction models based on annotated data



- ***Not a magical solution***: learning can be difficult *(e.g., twitter posts do not look like WSJ or NYT articles)*
  - ***Annotating data for high level tasks is difficult!***



Language Models

- A language model over a given vocabulary V assigns probabilities to strings drawn from V*

$$P_{ngram}(w_1 \ldots w_i) := P(w_1)P(w_2|w_1)\ldots P(\underbrace{w_i}_{nth\ word} | \underbrace{w_{i-n-1}\ldots w_{i-1}}_{prev.\ n-1\ words})$$

Unigram	$P(w_1)P(w_2)\ldots P(w_i)$
Bigram	$P(w_1)P(w_2\|w_1)\ldots P(w_i\|w_{i-1})$
Trigram	$P(w_1)P(w_2\|w_1)\ldots P(w_i\|w_{i-2}\ w_{i-1})$

Language Models

- A language model over a given vocabulary V assigns probabilities to strings drawn from V*

Our goal is to asses whether

P(Private Customer… Be Toad)

>?<

P(Private Customer… Be Towed)

Language Models

Unigram	$P(w_1)P(w_2)...P(w_i)$
Bigram	$P(w_1)P(w_2\|w_1)...P(w_i\|w_{i-1})$
Trigram	$P(w_1)P(w_2\|w_1)...P(w_i\|w_{i-2}\ w_{i-1})$

Our goal is to asses whether

P(Private Customer… Be Toad)

>?<

P(Private Customer… Be Towed)

What would be the answer if we use –
(1) a *Unigram* model? **(2)**a *Bigram* model?

Example: Trigram language model

- Consider the sentence:

Mr. Smith goes

$$p(\text{Mr. Smith goes STOP}) = p(\text{Mr.}|*,*))$$
$$p(\text{Smith}|*,\text{Mr.})\ \ p(\text{goes}|\text{Mr., Smith})\ \ p(\text{STOP}|\text{Smith, goes})$$

Model Estimation

- The last remaining issue – how can we estimate the models parameters $p(w_i|w_{i-2},w_{i-1})$
- Simple solution – *counting*!
  - *Also known as* ***Maximum likelihood estimate***

$$p(w_i|w_{i-2},w_{i-1}) = \frac{\text{count}(w_{i-2}, w_{i-1}, w_i)}{\text{count}(w_{i-2}, w_{i-1})}$$

How many parameters does the model need to estimate?

Model Estimation

- ***How many parameters does the model need to estimate?***
  - Let's assume a trigram model, defined over vocabulary V
  - The number of parameters is $\mathbf{|V|^3}$
  - Let's assume: |V| = 20K $< |V_{Shakespeare}|$
  - We'll have to estimate $\mathbf{8 \times 10^{12}}$ parameters
- **How many will we need to estimate for a unigram model?**
  - **Why not just do that?**

Generating from a distribution

- You can sample text from language models

unigram: Months the my and issue of year foreign new exchange's september were recession exchange new endorsed a acquire to six executives

bigram: Last December through the way to preserve the Hudson corporation N. B. E. C. Taylor would seem to complete the major central planners one point five percent of U. S. E. has already old M. X. corporation of living on information such as more frequently fishing to keep her

trigram: They also point to ninety nine point six billion dollars from two hundred four oh six three percent of the rates of interest stores as Mexico and Brazil on market conditions

Language Models

Unigram	• To him swallowed confess hear both. Which. Of save on trail for are ay device and rote life have • Every enter now severally so, let • Hill he late speaks; or! a more to leg less first you enter • Are where exeunt and sighs have rise excellency took of.. Sleep knave we. near; vile like
Bigram	• What means, sir. I confess she? then all sorts, he is trim, captain. •Why dost stand forth thy canopy, forsooth; he is this palpable hit the King Henry. Live king. Follow. •What we, hath got so she that I rest and sent to scold and nature bankrupt, nor the first gentleman? •Enter Menenius, if it so many good direction found'st thou art a strong upon command of fear not a liberal largess given away, Falstaff! Exeunt
Trigram	• Sweet prince, Falstaff shall die. Harry of Monmouth's grave. • This shall forbid it should be branded, if renown made it empty. • Indeed the duke; and had a very good friend. • Fly, and will rid me these news of price. Therefore the sadness of parting, as they say, 'tis done.
Quadrigram	• King Henry. What! I will go seek the traitor Gloucester. Exeunt some of the watch. A great banquet serv'd in; • Will you not tell me who I am? • It cannot be but so. • Indeed the short and the long. Marry, 'tis a noble Lepidus.

Generating Shakespeare

The Shakespeare corpus consists of N=884,647 word **tokens** and a vocabulary of V=29,066 word **types**

Only 30,000 word types occurred.
Any word that does not occur in the training data has zero probability!

Shakespeare produced 300,000 bigram types out of V^2= 844 million possible bigram types. 99.96% of the possible bigrams were never seen

Only 0.04% of all possible bigrams occurred. Any bigram that does not occur in the training data has zero probability!

Simple Solution: Smoothing (Add-1, Laplacian)

$$\text{MLE} \quad P(w_i) = \frac{C(w_i)}{\sum_j C(w_j)} = \frac{C(w_i)}{N}$$

$$\text{Add One} \quad P(w_i) = \frac{C(w_i)+\mathbf{1}}{\sum_j (C(w_j)+\mathbf{1})} = \frac{C(w_i)+\mathbf{1}}{N+\mathbf{V}}$$

A general tradeoff in ML, we will meet it again!

Perplexity

- Assume we have a language model
- Sample new (test) data for evaluation: $s_1,\ldots,s_m$
- We will look at the probability of the test data under out model: $\prod_{i=1}^{m} p(s_i)$
- Or, for convenience the log of that probability:

$$\log \prod_{i=1}^{m} p(s_i) = \sum_{i=1}^{m} \log p(s_i)$$

- **Perplexity is defined as:**

$$\text{Perplexity} = 2^{-l} \quad \text{where} \quad l = \frac{1}{M} \sum_{i=1}^{m} \log p(s_i)$$

(M is the number of words in the test data)

Perplexity

- Given a vocabulary V of size |V| = N
- We have a very "bad" model:

$$p(w|w_{i-2}, w_{i-1}) = 1/N$$

- What is the perplexity of this model?

$$\text{Perplexity} = 2^{-l} \quad \text{where} \quad l = \frac{1}{M}\sum_{i=1}^{m} \log p(s_i)$$

$$l = \log \frac{1}{N}$$

➔ The perplexity is N

Simple Intuition: Given the context what is *the effective "branching factor" for the current word.* ➔ Lower is better!

Evaluating Language Models

- **Which one is better** – unigram, bigram, trigram?
  - Can perplexity tell us that?
- Goodman 2001: |V| =50,000
  - **Trigram** model: Perplexity =74
  - **Bigram** model: Perplexity = 137
  - **Unigram** model: Perplexity = 955

- **Is a trigram model always better? (i.e., lower perplex)**

Rosetta Stone

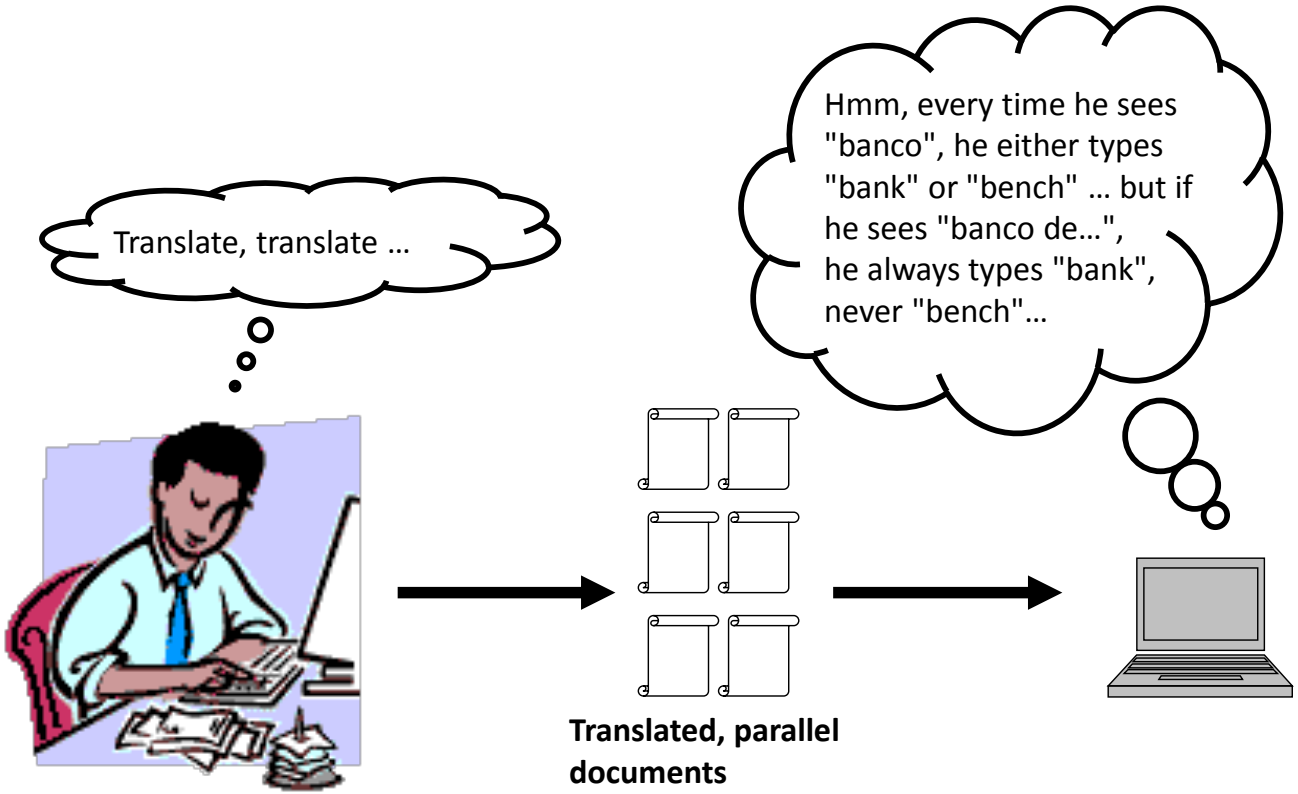
Dated to 196 BC, found in 1799.

The top register in Ancient Egyptian hieroglyphs, the second in the Egyptian Demotic script, and the third in Ancient Greek.

==> Parallel corpora!

Rosetta Stone proved to be the key to deciphering Egyptian hieroglyphs.

Statistical Machine Translation



Can you adapt the language model approach for translation?

Slides by K.Knight ACL'13

Let's try!

How do you say *strong* in Spanish?

1a. Garcia and associates . 1b. Garcia y asociados .	7a. the clients and the associates are enemies . 7b. los clients y los asociados son enemigos .
2a. Carlos Garcia has three associates . 2b. Carlos Garcia tiene tres asociados .	8a. the company has three groups . 8b. la empresa tiene tres grupos .
3a. his associates are not strong . 3b. sus asociados no son fuertes .	9a. its groups are in Europe . 9b. sus grupos estan en Europa .
4a. Garcia has a company also . 4b. Garcia tambien tiene una empresa .	10a. the modern groups sell strong pharmaceuticals . 10b. los grupos modernos venden medicinas fuertes .
5a. its clients are angry . 5b. sus clientes estan enfadados .	11a. the groups do not sell zenzanine . 11b. los grupos no venden zanzanina .
6a. the associates are also angry . 6b. los asociados tambien estan enfadados .	12a. the small groups are not modern . 12b. los grupos pequenos no son modernos .

Statistical Machine Translation

1a. Garcia and associates . 1b. Garcia y asociados .	7a. the clients and the associates are enemies . 7b. los clients y los asociados son enemigos .
2a. Carlos Garcia has three associates . 2b. Carlos Garcia tiene tres asociados .	8a. the company has three groups . 8b. la empresa tiene tres grupos .
3a. his associates are not strong . 3b. sus asociados no son fuertes .	9a. its groups are in Europe . 9b. sus grupos estan en Europa .
4a. Garcia has a company also . 4b. Garcia tambien tiene una empresa .	10a. the modern groups sell strong pharmaceuticals . 10b. los grupos modernos venden medicinas fuertes .
5a. its clients are angry . 5b. sus clientes estan enfadados .	11a. the groups do not sell zenzanine . 11b. los grupos no venden zanzanina .
6a. the associates are also angry . 6b. los asociados tambien estan enfadados .	12a. the small groups are not modern . 12b. los grupos pequenos no son modernos .

10,000 feet view:

$$P(W_{target\_i} \mid W_{source\_i-k}, \ldots, W_{source\_i})$$

Slides by K.Knight ACL'13

Text Decipherment

- Can we think about text decipherment as a special case of translation?
  - Assume the original text was “translated” into code.

```
PLAIN:   ABCDEFGHIJKLMNOPQRSTUVWXYZ
CIPHER:  PLOKMIJNUHBYGVTFCRDXESZAQW
```

  - Your job – translate it back!

Ciphertext: **NMYYT BUXXQ ...**

Slides by K.Knight ACL’13

Text Decipherment

Feels to easy to be true.. *What's wrong here?*

- So far we assumed –
  - Parallel corpora, allowing us to identify mapping between words
  - A ”dictionary” telling us how to translate (actual dictionary, or code book)

- But.. We typically don't get either one!
  - Essentially, this is an *unsupervised learning problem*

Text Decipherment

- *Is all hope lost?*

```
KDCY LQZKTLJKX CY MDBCYJQL: "TR

HYD FKXC, FQ MKX RLQQIQ HYDL

MKL DXCTW RDCDLQ JQMNKXTMB

PTBMYEQL K FKH CY LQZKTL TC."
```

Text Decipherment

If you know what is the underlying language you can make an educated guess.

KDCY LQZKTLJKX CY MDBCYJQL: "TR

HYD FKXC, FQ MKX RLQQIQ HYDL

MKL DXCTW RDCDLQ JQMNKXTMB

PTBMYEQL K FKH CY LQZKTL TC."

A	
B	3
C	8
D	7
E	1
F	3
G	
H	3
I	1
J	3
K	10
L	10
M	6
N	1
O	
P	1
Q	10
R	3
S	
T	7
U	
V	
W	1
X	5
Y	7
Z	2

Text Decipherment

```
a   o   e.a   .a    o      o.e      .
KDCY LQZKTLJKX CY MDBCYJQL: "TR

.o   .a     .e   a   . ee.e .o
HYD FKXC, FQ MKX RLQQIQ HYDL

 a         .  .     e .e .a
MKL DXCTW RDCDLQ JQMNKXTMB

.    o.e   a .a.   o   e.a
PTBMYEQL K FKH CY LQZKTL TC."
```

```
A
B 3
C 8
D 7  #
E 1  .
F 3  .
G
H 3  .
I 1  .
J 3  .
K 10 ##### V
L 10 ##
M 6  #
N 1  .
O
P 1  .
Q 10 ######### V
R 3  .
S
T 7  ### V
U
V
W 1  .
X 5
Y 6  #### V
Z 2  .
```

Slides by K.Knight ACL'13

Text Decipherment

a to repair.a to to.e if

KDCY LQZKTLJKX CY MDBCYJQL: "TR

.o .a t .e a freeze .o r

HYD FKXC, FQ MKX RLQQIQ HYDL

ar ti. f t re .e .a i

MKL DXCTW RDCDLQ JQMNKXTMB

.i o.er a .a. to repair it

PTBMYEQL K FKH CY LQZKTL TC."

frequent cipher letters: ~~Q~~ ~~L~~ ~~K~~ ~~C~~ D ~~T~~ M ~~Y~~ X
frequent English letters: ~~e~~ ~~t~~ ~~o~~ ~~a~~ n ~~i~~ ~~r~~ s h

```
A
B 3
C 8
D 7  #
E 1  .
F 3  .
G
H 3  .
I 1  .
J 3  .
K 10 ##### V
L 10 ##
M 6  #
N 1  .
O
P 1  .
Q 10 ######### V
R 3  .
S
T 7  ### V
U
V
W 1  .
X 5
Y 6  #### V
Z 2  .
```

Slides by K.Knight ACL'13

Text Decipherment

auto repairman to customer: if

KDCY LQZKTLJKX CY MDBCYJQL: "TR

you wait we can freeze your

HYD FKXC, FQ MKX RLQQIQ HYDL

car until future mechanics

MKL DXCTW RDCDLQ JQMNKXTMB

discover a way to repair it

PTBMYEQL K FKH CY LQZKTL TC."

```
A
B 3
C 8
D 7  #
E 1  .
F 3  .
G
H 3  .
I 1  .
J 3  .
K 10 ##### V
L 10 ##
M 6  #
N 1  .
O
P 1  .
Q 10 ######### V
R 3  .
S
T 7  ### V
U
V
W 1  .
X 5
Y 6  #### V
Z 2  .
```

Slides by K.Knight ACL'13

Text Decipherment

- That's starting to make more sense.
- If we know what is the underlying language
  - Distribution of characters, pairs of characters, triplets, etc.
  - Dictionary: provides a way to check if an interpretation is legal
  - Language model: Rank different interpretations as being likely or not.

Text Decipherment

- Just when you thought you got it…
- Our previous approach relied on
  - Mapping remains unchanged ..
  - ..code character distribution and text character distributions are the same
  - …character order is preserved

Enigma

Substitution system

N → J

Substitution table **changes** with every keystroke:

NNN → JTE

Flattens out ciphertext letter distributions.

Secret key = initial rotor ordering and settings

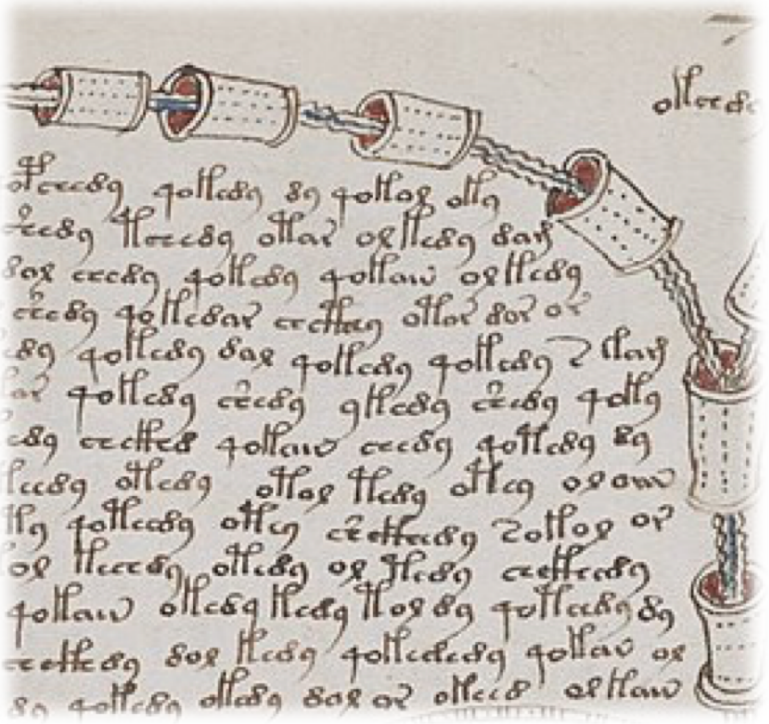
Reversible behavior

NNN → JTE → NNN

Text Decipherment

- **Quick summary**: we can think about decipherment as “translation”
  - **But**.. No codebook, no parallel corpora
- We assume non-parallel corpora
  - Compare character and word distributions over the two texts
- “Opponent” also knows it
  - They can hide the underlying language
  - They can they can reorder characters, change the mapping, etc.

Decipherment Challenge



The Voynich manuscript is written in an unknown script that encodes an unknown language

Decipherment Challenge

- *The Voynich manuscript is written in an unknown script that encodes an unknown language*
- The authors had to make some assumptions:
  - Symbols in scripts contain no more than a few dozen unique characters, which can modeled as substitution cipher.
  - Allow an unknown transposition scheme could have been applied to the enciphered text, resulting in arbitrary scrambling of letters within words (*anagramming*).
  - Consider the possibility that the underlying script is an abjad, in which only consonants are explicitly represented
    - Semitic languages

Decipherment Tasks

- The decryption system is composed of three steps.
  - **The first task** is to *identify the language of a cipher-text*, by comparing it to samples representing known languages.
  - **The second task** is to map each symbol of the cipher text to the corresponding letter in the identified language.
  - **The third task** is to decode the resulting anagrams into readable text, which may involve the recovery of unwritten vowels.

Source Language Identification

- Given a text, and a set of candidate languages, how can you tell what language is the text in?

- Authors suggest three methods:
  - *Relative character frequencies*
  - *Patterns of repeated symbols*
  - *Outcome of a trial decipherment*

Character Frequency

- **Simple approach**: guess the source language of a cipher text is by character frequency analysis.
  - *The relative frequencies of symbols in the text are **unchanged after encipherment** with a 1-to-1 substitution cipher.*
- **Algorithm**:
  - Order the cipher text symbols by frequency, normalize these frequencies to create a probability distribution
  - Choose the closest matching distribution from the set of candidate languages.

Character Frequency

The probability of the ith character in a text U

$$d(U,V) = -\ln \sum_i \sqrt{P_U(i) \cdot P_V(i)}$$

Strengths:

Robust to letter reordering

Lack of word boundaries

Patterns of Repeated Symbols

- Expands on the character frequency method by incorporating the notion of *decomposition patterns*
- This method uses multiple occurrences of individual symbols within a word as a clue to the language of the cipher text.
  - For example, the word *seems* contains two instances of ‘s’ and ‘e’, and one instance of ‘m’ ==> replace with (2,2,1)
- Captures the relative frequency of such patterns in texts, independent of the symbols used.
  - Compute distances over the distributions of these patterns

Trial Decipherment using Character Language Models

- **Key idea:**
  - Construct a character level language model for a candidate language.
  - Find the decipherment that maximizes the probability of the interpreted text, based on the language model
- Greedy search over possible decipherment options
- Pick the language that has the **highest best** decipherment

Language Identification Evaluation

- The Universal Declaration of Human Rights(UDHR) in 380 languages
- Divided the text in each language into 66% training, 17% development, and 17% test.

Method	Dev	Test
Random Selection	0.3	0.3
Jaskiewicz (2011)	54.2	47.6
Character Frequency	72.4	67.9
Decomposition Pattern	90.5	85.5
Trial Decipherment	94.2	97.1
Oracle Decipherment	98.2	98.4

Table 1: Language identification accuracy (in % correct) on ciphers representing 380 languages.

Voynich Experiments: Language identification

- Contains 17,597 words and 95,465 characters
- Candidate languages:

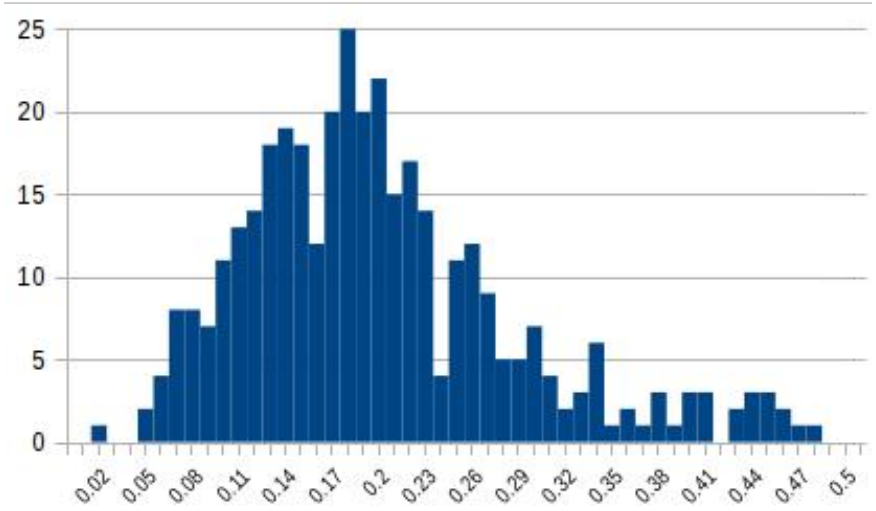
Language	Text	Words	Characters
English	Bible	804,875	4,097,508
Italian	Bible	758,854	4,246,663
Latin	Bible	650,232	4,150,533
Hebrew	Tanach	309,934	1,562,591
Arabic	Quran	78,245	411,082

Table 4: Language corpora.

- Note the choice of old text, to best resemble the Voynich text
  - Don’t try this with Twitter..

Voynich Experiments: Language identification

- Results for the **pattern decomposition**: very strong resemblance between Hebrew and the Voynich text!



- **Trial decipherment** approach: Hebrew and Esperanto take the top places
  - Esperanto is an older language.

Script Decipherment

- After identifying the language – decipher the text!
  - Reverse the substitution cipher
  - Unscramble the anagram words into readable text

(a) organized compositions through improvisational music into genres
(b) fyovicstu dfnrfecpcfie pbyfzob cnryfgcevpcfivm nzecd cipf otiyte
(c) otvfusyci cpifenfercfd bopbfzy fgyiemcpfcvrcnv nczed fpic etotyi
(d) adegiknor ciimnooopsst ghhortu aaiiilmnooprstv cimsu inot eegnrs
(e) adegiknor compositions through aaiiilmnooprstv music into greens

Figure 3: An example of the encryption and decryption process: (a) plaintext; (b) after applying a substitution cipher; (c) ciphertext after random anagramming; (d) after substitution decipherment (in the alphagram representation); (e) final decipherment after anagram decoding (errors are underlined).

Script Decipherment

- *Reverse the substitution cipher*
  - **Compute the score** for each key using a character level and world level language model.
- *Anagram Decoder*
  - Right letters, wrong order
  - Convert to alphagrams, find most likely word sequence based on a word-level language model
    - E.g., the alphagram *flow* corresponds to the words flow and wolf. `P(water flow) > P(water wolf)`

Voynich Experiments: Script Decipherment

- Decipher first 10 pages using the 5 language models
- Evaluate based on in-vocabulary items

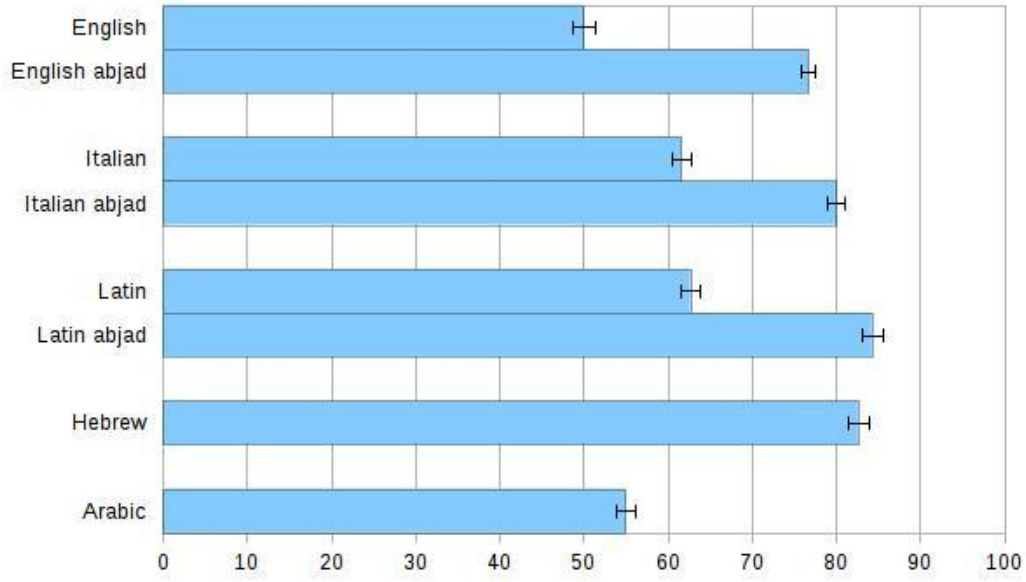
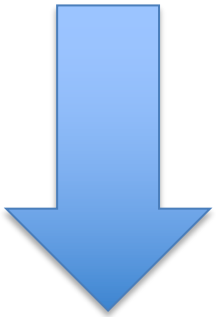


Figure 6: Average percentage of in-vocabulary words in the decipherments of the first ten pages of the VMS.

Voynich Experiments: Script Decipherment

ועשה לה הכהן איש אליו לביחו ו עלי אנשיו המצות



Some spelling corrections +
Google translate

She made recommendations to the priest, man of the house and me and people

Voynich Experiments: Script Decipherment

Unigram assessment based on topic matching.
For example these words were identified in the herbal section -

הצר ‘*narrow*’ **איכר** ‘*farmer*’

אור ‘*light*’, **אויר** ‘*air*’, **אש** ‘*fire*’.