暗号・情報保全史特論
History of Cryptograph and Signal Security Advanced Course

第3・4回: 近世～第2次世界大戦期の暗号発達史と暗号解析の技法

佐藤永欣

16世紀以降のヨーロッパの暗号

- 古くからの暗号の強度が怪しくなる
  - シーザー暗号
  - 単一換字暗号
- 多表式暗号が考案される
  - ヴィジュネル暗号(前回やった)
  - ヴィジュネル暗号の変種の時代が続く
    - どの行を使うかを選択する手法の変更とかそのレベル
  - ヴィジュネル暗号の換字表をシーザー暗号式から単一換字暗号に変える
    - このころになると機械式暗号が視野に入る

頻度分析

- 平文の特徴を使った暗号解読法の一つ
- 大量に暗号文を集める→文字の出現頻度を数える→出現頻度が高いはずの文字を当てはめて文が成り立つか検討
  - 概ね母音字、次に発音しやすい子音字、あまり使わない子音字の順
- 9世紀ごろまでにアラビアで考案される→15世紀ごろまでにヨーロッパに伝わる
  - (˘ω˘).oO(ところでアラビア文字には母音が無いんだよなあ)
    - 正確には表記法はあるが表記しない。そもそも母音がi, u, aの3種類+長短+ai,auの組み合わせだけ(eとoはあるが外来語のみ)
    - どうやって頻度分析に気づいたんだ???
    - 子音が28種類。子音だけの表記体系で、逆に気づきやすかったのかもしれない??

頻度分析

- 自然言語の書き言葉における統計的な文字の出現頻度(英語)

文字	頻度[%]	文字	頻度[%]	文字	頻度[%]
A	8.19	J	0.14	S	6.36
B	1.47	K	0.41	T	9.41
C	3.83	L	3.77	U	2.58
D	3.91	M	3.34	V	1.09
E	12.25	N	7.06	W	1.59
F	2.26	O	7.26	X	0.21
G	1.71	P	2.89	Y	1.58
H	4.57	Q	0.09	Z	0.08
I	7.10	R	6.85		

出典: 長田順行, “暗号大全 原理とその世界”, p.127, ニューヨークタイムズ10万字統計

頻度分析

- 頻度分析が有効な暗号
  - シーザー暗号
  - 単一換字暗号のうち1:1変換を行っているもの
- 頻度分析に対抗して考え出されたと思われる暗号
  - 単一換字暗号のうち1:多変換を行っているもの(度数秘匿式)
    - 出現頻度の多い文字を数種類の暗号文字に変換する方式
    - 出現頻度の少ない文字について多:1変換をするものをふくむ
  - 多:多変換を行う暗号のうち、換字表を順次切り替えないもの
    - th, quのような頻出綴りを文字とみなして暗号文字に変換する方式
  - 基本的に、統計的に有意な頻度の差が無くなるように

頻度分析の例

- 推理小説の例がわかりやすい?
  - 青空文庫にあるので読んで。たぶん図書館にもある(投げやり
  - 黄金虫(エドガー・アラン・ポー): 数字と記号列
    - https://www.aozora.gr.jp/cards/000094/card2525.html
  - 踊る人形(コナン・ドイル): UMLの人型みたいな落書きの列
    - https://www.aozora.gr.jp/cards/000009/card50713.html
- 多分踊る人形の方が読みやすいかな?

頻度分析の考え方の拡張

- 複数文字の連続パターン: 英語に限らず、平文に頻出する綴りのパターンがある。
  - ２文字の場合
    - th: ローマ字にない子音を表現するために組み合わせ(英語にはあるがラテン語・イタリア語にはない。ドイツ語にもない)
    - qu: ラテン語の代名詞(qui, quis, quisque)由来?(たぶん。Questionとかquestとか)
    - gh: 古英語・中英語では発音されていた子音由来のつづりが印刷術の普及で固定
    - まだあるはず。
  - ３文字の場合: 2文字の場合+母音など
    - sch: ドイツ語では頻繁に現れる
- 当然カナでも同じようなことは起こる
  - 日本語の場合は決まった言い回し由来が多い?

頻度分析の考え方の拡張

- ２文字・３文字…のハミング距離
  - t→h: 14文字(-12文字)
  - q→u: 4文字
  - g→h: 1文字
- シーザー暗号では同じハミング距離が現れる
- 単一換字暗号では3文字以上のハミング距離の場合にn→m文字のようなパターンが現れる
- 多表式暗号では斜行特性の判定などに利用できる

クリブ (crib)

- 暗号文に現れる同じ文字列
  - 下線を引いたところと*斜体*にしたところがそれぞれクリブ(シーザー暗号の例)
  - BXSMJDPMDA XU AWM *RMRXQM* VT AWM *RMXRQM* UXJ AWM *RMXRQM*
  - government of the people by the people for the people
- シーザー暗号・単一換字暗号では平文の単語がそのままクリブとして表れる
- 多表式暗号では平文中の同じ位置or多表の繰り返し周期のn倍の位置にある語がクリブとして現れる
  - 多表のうちの1枚の繰り返し周期でも十分
- 多表式暗号のうち表が周期的に切り替わるものはGTR→VIGのような隣接する文字のハミング距離にクリブのようなものも現れる(G→T=V→I=13, T→R=I→G=-2)

クリブの拡張

- ハミング距離への拡張(ハミング距離の定式化は戦後)
  - 多表式暗号に対抗するために現れた
    - 戦前の日本陸軍ではハミング距離を字差と呼んでいた。ストリップ暗号を解読するのに便利だったという

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**	**n**	**o**	**p**	**q**	**r**	**s**	**t**	**u**	**v**	**w**	**x**	**y**	**z**
0	E	K	M	F	L	G	D	Q	V	Z	N	T	O	W	Y	H	X	U	S	P	A	I	B	R	C	J
1	J	E	K	M	F	L	G	D	Q	V	Z	N	T	O	W	Y	H	X	U	S	P	A	I	B	R	C
2	C	J	E	K	M	F	L	G	D	Q	V	Z	N	T	O	W	Y	H	X	U	S	P	A	I	B	R
3	R	C	J	E	K	M	F	L	G	D	Q	V	Z	N	T	O	W	Y	H	X	U	S	P	A	I	B

- the→pdm (14,23→15,10)
  - 多表の変化が少ない表は同じと考えてここでは無視
  - 表の周期ごとに同じハミング距離が現れる

ラテン方陣

- いるかな????
- n×nの正方行列
- 正方行列の要素はすべて異なる
- 正方行列を送受信双方で共有し、座標を伝達
- 5×5行列だと頻度分析に負けてしまう
  - 10×10ぐらいにして出現頻度の高い文字をE1, E2, E3, …のように割り当て
  - 秘匿度数式につながる発想

秘匿度数式・逆秘匿度数式

- 頻度分析に対抗するために考案された過渡的な方法
- 頻度分析により、頻出する自然言語の文字が特定されないように:
  - 26文字のアルファベットを26よりも十分多い種類の記号に置換
  - Eのような頻出する文字を複数の記号に変換
  - すべての記号の出現頻度が概ね等しくなるように配慮して暗号化
- 逆度数秘匿式
  - 26文字のアルファベットを26よりも少ない種類の記号に変換
  - Z、XやQのような出現頻度の小さい文字を一つの記号に統合
  - 出現頻度が高い母音字を割と出現頻度高めの子音字と統合
  - 10種類程度の記号に変換する方式をローマ教皇庁が使っていた模様
  - 復号化する際には綴りを考慮してどの文字なのかを決定

綴り字換字式

- 平文の2文字一組で暗号文の記号に対応させる
  - 一応は頻度分析への対抗になる: 単純計算で26倍の通信文が必要
    - 暗号文の記号には数字も入れていい

↱	a	b	c	d	e	f	g	h	…	z
a	JE	XA	BQ	LM	QF	LA	AQ	XW		TV
B	KS	MD	VW	SF	GT	UB	LP	HJ		YD
c	IJ	HG	FC	RS	OA	PZ	VS	ET		OG
d	XP	LK	MA	RG	FJ	NH	JL	SQ		MM
e	OS	AF	KL	BW	LH	UA	NW	AA		AI
f	AC	VH	US	AS	XT	JJ	GD	JT		IZ
g	HP	TZ	JW	FO	RU	AK	EX	MQ		ZA
h	WI	VU	QK	KK	LV	JF	UU	BQ		HC
…										
z	EC	DW	SN	BA	CZ	KN	XO	PN		ES

プレイフェア暗号

- ホイットストーンが発明: なぜか友人の名前がついている
  - 綴り字換字式よりは暗号強度が低い

1. 平文を2文字ずつ区切って2文字組を作る
   - 同じ文字の組になった場合XかZを挿入して別の組にする
   - 1文字だけになった場合も1文字追加して2文字組にする
2. 2文字組の表中の位置により次の場合に分けて表を引く
   - 異なる行・列にある場合: 2文字を含む四角形の対称位置にある2文字を選ぶ
   - 同じ列にある場合: それぞれ一つ下の文字を選ぶ。一番下の場合は一番上を選ぶ
   - 同じ行にある場合: それぞれ一つ右の文字を選ぶ。一番右の場合は一番左を選ぶ

プレイフェア暗号

- 非常に単純なプレイフェア暗号表の例
  - 実際にはランダムにする。暗号表も頻繁に更新
- 暗号化例
  - au→EQ、th→IS (四角形の対称位置を選ぶ場合)
  - mb→RG、pu→UZ (同じ列)
  - ru→SQ、be→CA (同じ行)

A	B	C	D	E
F	G	H	I/J	K
L	M	N	O	P
Q	R	S	T	U
V	W	X	Y	Z

auの例

a	B	C	D	**E**
F	G	H	I/J	K
L	M	N	O	P
Q	R	S	T	**u**
V	W	X	Y	Z

コードブックの時代

- コードブックについて
  - 1冊制
    - 平文と暗号文の対応が割と規則的: 1冊あれば逆引きも正引きもできる
      - ABABA:a, ABACE: abandoned, ABADI: abated , ABECA: abided … ZPASI: zoo
      - 暗号語の近傍を探せば平文の語が載っている
  - 2冊制
    - 平文と暗号文の対応が規則的ではない。暗号化用と復号化用の2冊セットがないと作業しにくい
      - 古いものはランダム性に気を使っていない
- 秘匿のほかに文字数削減(=電報料金が安くなる)も目的とされた。商業用の暗号は暗号書が市販されていた
  - いろは引電信暗號, 明治28年7月
  - ACME Commodity and Phrase Code, 1923
- 秘匿目的のコードブックでも現代的な意味での暗号強度には寄与しない
- とはいえ、シンプルでわかりやすいので電信が普及しはじめた時代には暗号として多用された

コードブックの時代

- 16世紀後半以降のヨーロッパでは:
  - Cypherによる暗号化よりもCodeによる暗号化が好まれた
    - ルイ14世の大暗号もコードブック。音節単位の変換も併用
  - Cypherでは手作業による暗号化と復号化がかなり面倒
  - コードなら単語単位なので割と簡単・早い
- 暗号名・秘匿名称・筆名: ある意味コードの一種
  - コルネット作戦・オリンピック作戦: 沖縄占領後の日本本土上陸作戦の名称
  - オーバーロード作戦: ノルマンディ上陸→ドイツ方面侵攻の作戦名
  - スターリン: 本名ジュガシビリ。筆名が一般化して苗字のように使われている。「鋼鉄の人」という意味らしい
  - レーニン: 本名ウリヤノフ。共産党非合法時代の秘匿名称・筆名が一般化。「レナ川の人」という意味らしい

コードブックの時代

- 主に単語ベースでコードを決める
  - コードブックを作成・利用している組織の主な関心事がコードに反映される
    - 軍なら軍事関係の用語と想定している作戦地域の地名、対立する軍の人名
      - 蒋介石、張作霖などの主要な人名がコードされていた
    - 外交用コードブックなら相手国関連の地名・人名・経済関係
  - 関心事関連の単語は細かい変化形まで別コードのことが多い
  - 頻出する単語には複数のコードを割り当てる→出現頻度を分散
- 文字1文字へのコード割り当てもある
  - 単語の変化形を作るのに使われる
  - ありがちな語尾等は別に割り当て: -ed, -ing, -tion等
  - Cypher出身の暗号兵が手抜きをして文字1文字のコードで暗号化して解読の手掛かりを与えた事故もあった

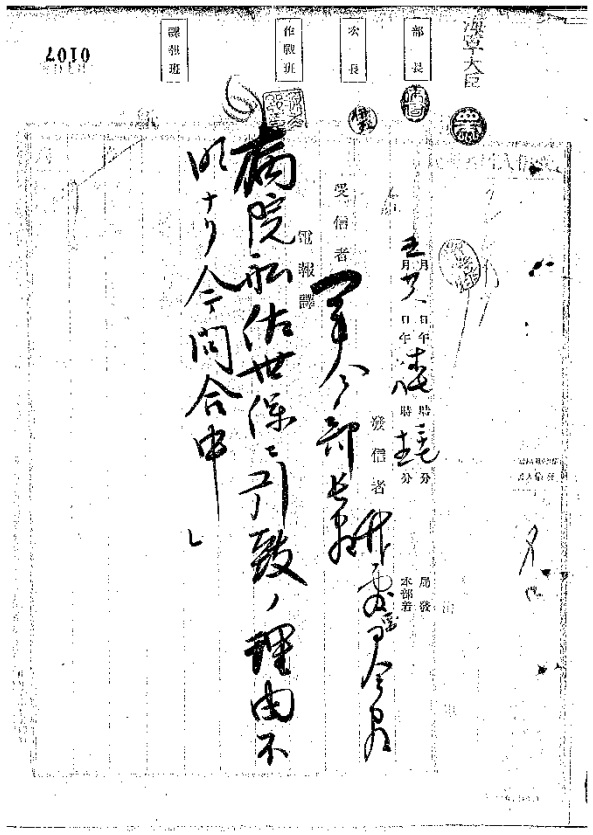
コードブックの時代

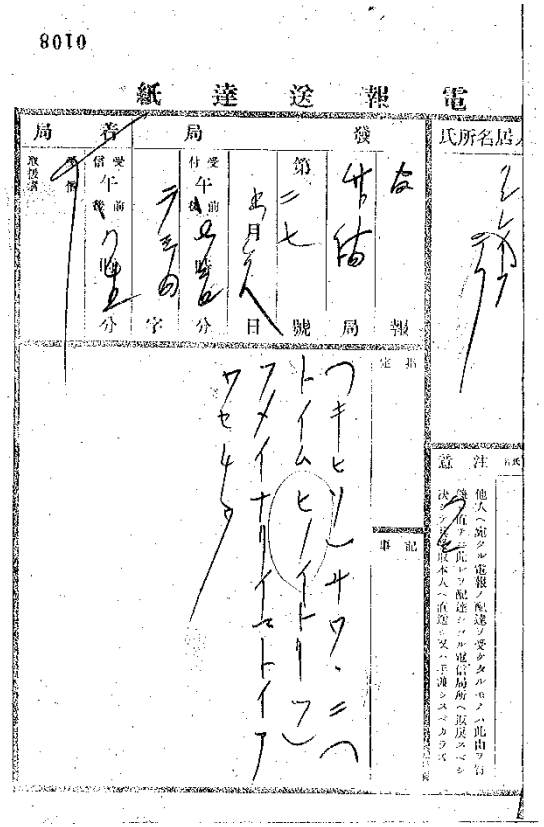
- 日露戦争～第1次世界大戦ごろまでは、暗号の秘匿はほぼコードブックに頼っていた
- 電文の例: 日露戦争(日本海海戦)
  - ヒヒヒ「YR」
  - 敵対馬東水道ヲ通過セントスルモノノ如シ
  - タタタタ　（モ四五六）「YR」セ
  - 敵艦隊見ユ　四五六地点　信濃丸
  - （アテヨイカヌ）ミユトノケイノウニセツシ（ノレツヲハイ）タダチニ（ヨシス）コレヲ（ワケフウメル）セントスホンジツテンキセイロウナレドモナミタカシ
  - 敵艦隊見ユトノ警報ニ接シ　聯合艦隊ハ　直ニ出動　之ヲ撃沈滅セントス　本日天気晴朗ナレ共波高シ
    - 平文とコードが混ざっている
    - 平文がコード解読の糸口を与えるが、当時知られていなかったor気にしていなかった

コードブックの時代

- 日露戦争の続き
  - アテヨ ニケヤ クニイ マタフ ニシヤ
  - 敵現出ニ付第三艦隊出港
  - 「h5rv00」メ「v」モ三〇七「at」ヨリ「K」アテ
  - 敵艦隊変針北東地点三〇七　笠置ヨリ三笠アテ
  - (キヒソ)サワニ(トイムヒノイトリツ)フメイナリイマトイアワセチウ
  - 病院船佐世保ニ引致の理由不明ナリ今問合中
  - シナノマルヨリサ　「RON」セ　ヲ　ホカクス　モ四二八
  - 信濃丸より(佐世保)　我レ「シソイベリキ」ヲ捕獲ス　地点四二八
  - ママナヨリ　セ「ROt」チンボツセリ
  - 満洲丸ヨリ　「アドミラル・ナヒモフ」沈没セリ
  - サドマルヨリ　セ「ROW」モ四二六　セントウリヨクナシ　ホカクス
  - 佐渡丸ヨリ　「ウラジミル・モノマフ」四二六地点　戦闘力ナシ　捕獲ス

コードブックの時代





電報綴り
左: 復号化後の清書(これでもまだ読み取れるほう)
右: 通信手がモールス信号を受信した用紙

(キヒソ)サワニ(トイムヒノイトリツ)フメイナリイマ
トイアワセチウ
病院船佐世保ニ引致の理由不明ナリ今問合中
ヒキソ:病院船、トイム: 引致、ヒノイ: の、トリツ: 理由
サワ: 佐世保(軍港名は2文字の略語だったらしい)

電報綴りは防衛庁防衛研究所が所蔵、国立公文書館アジア歴史資料センターが公開。
JACAR Ref. C09050518300「日本海海戦　電報報告１　明治３８・５・２７」
JACAR Ref. C09050519400「日本海海戦　電報報告２　明治３８・５・２８以降」

国立公文書館 アジア歴史資料センター
Japan Center for Asian Historical Records
http://www.jacar.go.jp

コードブックの解析

- 平文との対応を見てみる
  - アテヨイカヌ: 敵艦隊
  - ノレツヲハイ: 聯合艦隊ハ
  - ヨシス:出動
  - ワケフウメル:撃沈滅
  - キヒソ: 病院船
  - トイム: 引致
  - ヒノイ: の
  - トリツ: 理由
  - モ+漢数字三桁: 地点コード
  - 艦船・陸地の名称: アルファベット1～3文字

コードブックの解析

- コードの使用率によって難易度が変わる
  - 平文とコードが混用されている電文: 国語のテストの穴埋め問題のようになる
  - 全部コードの電文: 電文先頭の定型部分etcを切り出す
- コード部分が判明したら、
  - コードの頻度分析
  - 文法的な意味解析(語順・係り受けの関係etc.)→コードの品詞の特定
  - 名詞
    - 固有名詞: 関心があるはずの地名・艦船名・人名と発局・宛局から推定
    - 作戦区域や先頭の結果から事後的に推定できる
  - 動詞: 名詞と大体同様。そもそも種類が多くない
- 事後的な解析でもコードブックが更新されない限り有効

日露戦争当時の通信

- 海底ケーブルによる有線電信
  - 大西洋横断ケーブル・ロンドン-上海間のケーブルがあった
    - ロンドン-上海が直結されているわけではない。中継・転送を繰り返して電報が届いた
  - 上海-日本本土間の海底ケーブル
  - 各地の鎮守府・要港部・海軍望楼の間の電信: これも中継を繰り返していた
- 無線電信
  - 通信可能な距離が200キロ程度まで。受信機の能力が低かった
  - 他の無線局による中継や有線電信による中継を組み合わせていた
  - なお、陸軍は伝書鳩も使っていた(というより伝令将校と伝書鳩に頼っていた)
- 当時の電報綴りなどを見ると秘匿に対する意識が低かった模様
  - 主にコードによる暗号化
  - 艦名コードなどにかなとアルファベット2種類の体系がある(有線と無線での使い分けとも思われない)
  - 平文でも送っている

解読しやすそうな暗号文を作らせるテクニック(?)

- 暗号システムに対する攻撃の動機→敵対しているから
  - 解読したい暗号システム上では他者に関する情報もやり取りしている
  - 他者: 大抵は最大の関心事である敵対者のこと
  - つまり、敵対者が自分たちの情報を(うまいことして)流させることがある
- 主な目的
  - 秘匿されている名称・コードなどを明らかにする: 敵に有利になりそうな情報etcを流すように仕向ける
    - コードブックにより秘匿されている固有名詞を送信させる
    - 新奇な名詞を頻用させコードを追加させる
  - 既知の平文を送信させ暗号鍵(の一部)を明らかにする
    - 新聞記事や外交公文: 通常はパラフレーズされるが

解読しやすそうな暗号文を作らせるテクニック(?)

- 歴史上の例: ミッドウェイ海戦前
  - 旧日本海軍は太平洋の島や要地を表すのにアルファベット2文字の地名コードを使っていた
    - コードブックは頻繁ではないものの更新されていた&新しく作戦地域になるとコードが頻出する(作戦地域になるまでは該当するコードほぼ使われないから知られない)
    - コードで暗号化した通信文をさらにcypherで暗号化していた
  - AFという地名コードが捕捉された
    - 珊瑚海海戦でアメリカの空母を全部沈めたと日本が誤認→ハワイから西で自由に行動できると考えた→次の侵攻作戦
  - 共起関係にあるコードが港湾や飛行場の意味だとわかっていた
  - 条件がそろうのはミッドウェイ島だが確信が持てない
  - そこでミッドウェイでは海水蒸留機が故障したと偽情報を流した
    - AFでは真水が不足と日本側が報告する電文を送信
    - 実はどこまで本当なのかわからない: 一次情報が見当たらない

補足: 無線電信の歴史

- 1895年ごろ～
  - マルコーニによる無線電信の実用化
  - 当初は火花式無線電信だった
    - 一定周波数を発振できる回路がそもそもなかった
    - コイルに高電圧をかけておく→回路を切断→火花が飛ぶ→火花がアンテナから電波として放射される→数ヘルツ～数メガヘルツの電波を発射
    - 周波数無関係に、電波が届く範囲の通信は全て混信した
      - 艦載無線機の通信可能距離が100～200km程度だった模様
      - 同調回路の特許が1900年、真空管による増幅回路はまだない
  - 日露戦争当時、火花式無線機を国産化して艦艇に搭載
    - ロシアも艦艇に搭載していたが重要性を理解できていなかった。日本側の通信を妨害しなかった
    - タイタニックも火花式無線電信

補足: 無線電信の歴史

- 1900年ごろ
  - 同調回路が使われ始める→特定の周波数だけの電波を発射
    - 初期のころは増幅回路がなかった。同調による共振だけで信号をピックアップしていた
    - 真空管の発明が1904～1906年。原理となるエジソン効果の発見が1884年
  - 搬送波の断続でモールス信号を送るようになった
  - 短波帯の利用→電離層反射をつかった長距離無線通信
- 1920年ごろ～
  - ラジオ放送開始
  - 出力が小さい割に遠距離を通信できるので第2次世界大戦中も電信が使われた
    - 無線電話は送受信機が大きくなりすぎた

ワンタイムパッド

- 解読不能（絶対安全）なことが数理的に証明: いくつかの条件が必要
  - Claude Shannon, "Communication Theory of Secrecy Systems", *Bell System Technical Journal*, Vol. 28, No. 4, pp. 656-715. (1949)
  - シャノンによる証明以前から解読の糸口が皆無であることは知られていた→安全とされていた
- 絶対安全を実現するために
  - 平文と同じ長さの鍵を用意する → 平文より短い鍵は鍵の再使用と等価
  - 鍵は乱数で生成する
  - 鍵は再使用しない → 再使用すると解読され得る
- 平文を鍵で暗号化(暗号化の計算自体は下記に限らない)
  - 平文 XOR 鍵の計算をする（鍵長が平文より短いことを許容するとバーナム暗号。ただし、安全ではない）
  - 加算してMod 26する
- 暗号文を送る
- 鍵を「安全な手段で」送る（かなり難しい）
- 「安全な手段」が使えない/使いにくいから平文を暗号化して送ろうとしているはず…
  - スパイ、前線部隊の通信に使われた。連隊以下の規模の部隊?
  - 連隊規模以下なら直接行き来しやすい

ワンタイムパッドの問題点

- 乱数表の作成・配布がすごく大変
- 放送方式の命令伝達に使いにくい
  - おなじ乱数表を複数の受信者に配布?
    - 乱数表が鹵獲される恐れがある
    - 通信相手によって乱数表使用の進度が異なる
    - 異なる通信相手が送信すると乱数表の繰り返し使用と同じ
- 電文をリレーする場合も大変
  - 大本営(東京)→支那派遣軍(南京)→第6方面軍(漢口)→第20軍(衡陽)→第64師団(長沙)→歩兵69・70旅団→歩兵大隊各4個大隊
    - 暗号化と復号化を繰り返さないといけない
    - 日本陸軍の場合は中枢の通信には使われなかった
    - 師団レベル以下では使われていた←配布は伝令を使える
- 乱数表をすごい勢いで使用してしまう
  - 1日200通ぐらい通信する→1通当たり200語としても4万／日消費

課題用プログラムなど

- 基本的にrubyで書いたスクリプト
  - FreeBSD上のruby 2.7.5と3.1.2と3.2.0で動作確認
  - 文字列をゴショゴショやる以外に変なことはしていないのでLinuxやWindows上のrubyでも動くはず
  - 必要なライブラリも特にない(はず)
  - Windows用のrubyは https://rubyinstaller.org/ あたりから持ってくる
  - ソフトウェア演習で(今も?)使うLinux環境でも(rubyが入っていれば)動くはず
- GUIはないので頑張って
- Enigmaやフリードマンテストのプログラムもあるが、これらが出てくる回で解説する
  - 今回は頻度分析と単一換字暗号だけ

課題用プログラムなど

- 頻度分析
  - count_freq.rb
  - 標準入力に含まれるアルファベット26文字分の出現頻度をカウント。記号や数字は無視。日本語も無視
- 単一換字暗号
  - 換字表生成プログラム: gen_key_simple_subst.rb
  - 空白区切りの単一換字表を生成する
    - 平文のa, b, c, …, zに対応する暗号文の文字が26文字分、1行に入る仕様
  - encrypt_simple_subst.rb: 単一換字暗号の暗号化
  - decrypt_simple_subst.rb: 単一換字暗号の復号化
  - 使い方: encrypt_simple_subst.rb キーファイル
    - 標準入力から平文/暗号文のテキストを入力、標準出力に暗号文/平文のテキストを出力
    - アルファベット26文字の事しか考えていないので、算用数字はそのまま
    - 日本語を食わせたらどうなるかは未確認