情報漏洩を防止するための Word2Vecを用いた 機密情報に対する推論規則生成

木下研究室　加来 功平

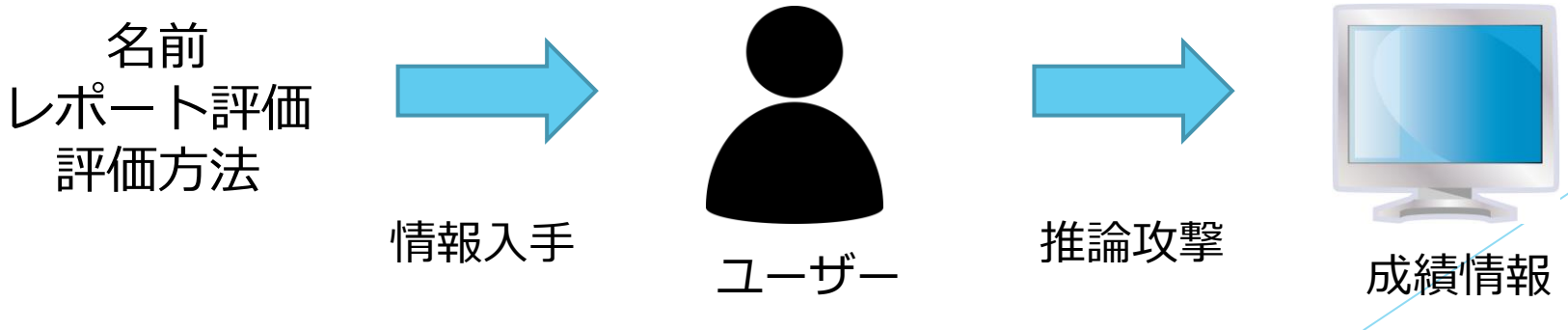
研究背景

近年、ownCloudやiCloudなどのクラウドサービスが開始し扱う情報量が増えていく中で、情報漏洩のリスクも高くなってきている。

問題点

公開されている複数の情報から、秘密されている情報が推論される可能性がある。

～推論攻撃の例～



【目的】推論規則を自動生成する

1. 機密テクストを公開テクストから推論されないようにしたい。
2. そのために推論規則を自動生成する。
3. 自動生成にWord2vecを使用する。
4. Word2vecは単語やテクストの間の確率的関連が計算できる。
5. 「テクストの確率的関連性」と「人間が理解する意味の関連性」は相関があるのだろうか？
6. 実験してみる。

【提案】PMMI

- PMMI(Pointwise Mutual Multiple Information)は3つの事象の間の関連度合いを測る尺度とする。ただし3つの事象は独立であると仮定する。定義式を以下に示す。

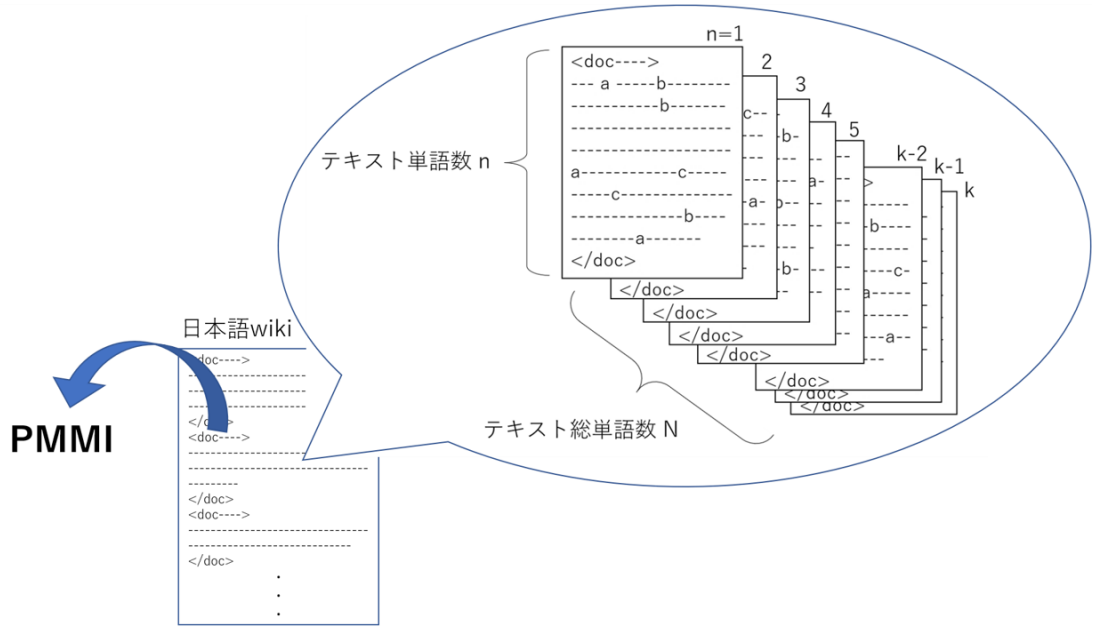
$$\mathrm{PMMI}(x, y, z) = \log_2 \frac{p(x,y,z)}{p(x)p(y)p(z)}$$

$$= \log_2 p(x,y,z) - \log_2 p(x)p(y)p(z)$$

【提案】PMMI

日本語wikiテキスト(3.3GB)のPMMI

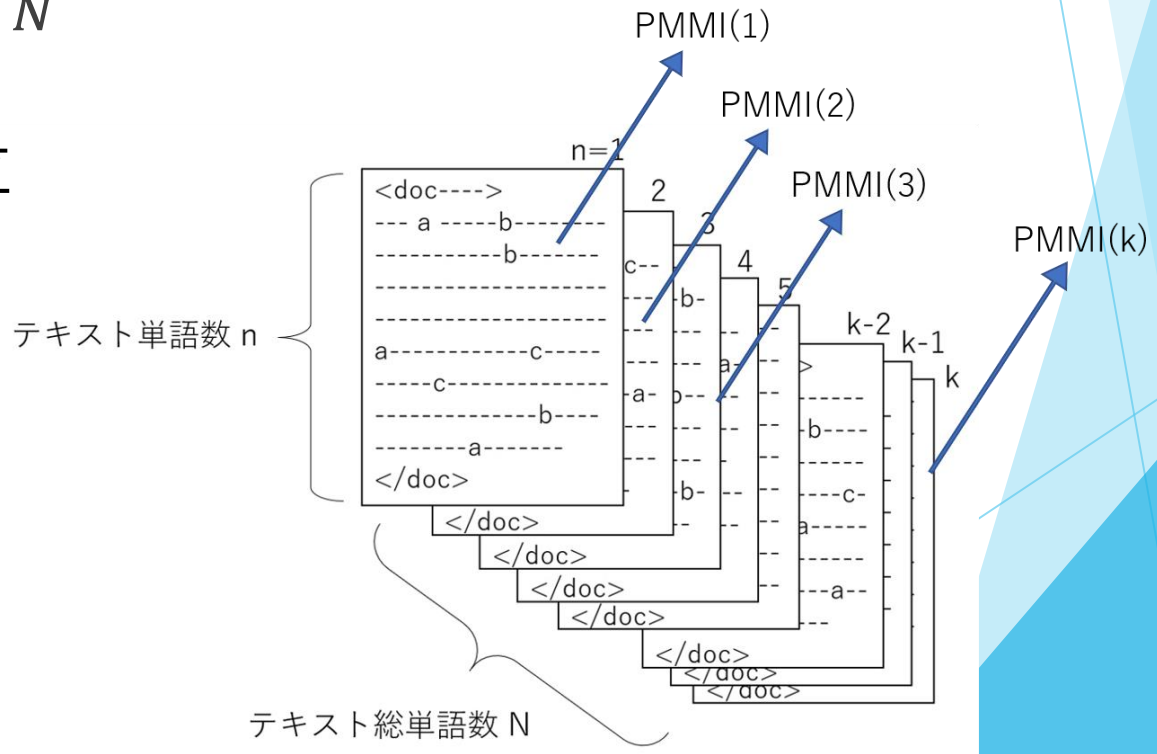
- PMMI= $\log_2 \frac{p(x,y,z)}{p(x)p(y)p(z)} = \log_2 \frac{\prod_{i=1}^{k}(\frac{a}{n}) \prod_{i=1}^{k}(\frac{b}{n}) \prod_{i=1}^{k}(\frac{c}{n})}{\frac{A}{N}*\frac{B}{N}*\frac{C}{N}}$



【提案】 PMMI

- $\mathrm{PMMI(k)}=\log_2 \frac{P(a)*P(b)*P(c)}{\frac{A}{N}*\frac{B}{N}*\frac{C}{N}}$

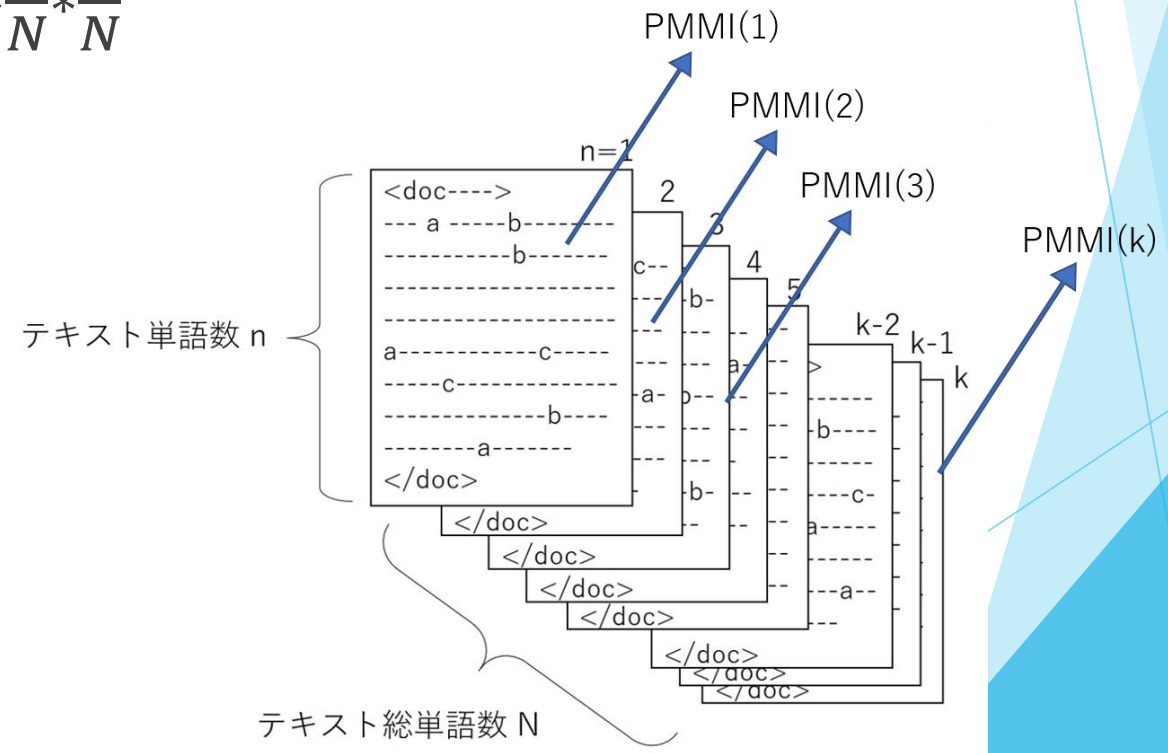
単語 a ,b ,c はそれぞれ独立であるため、同時分布確率はそれぞれの単語の出現頻度を掛け合わせることにより求められる。



【提案】PMMI

- $\mathrm{PMMI(k)}=\log_2 \frac{P(a)^a * P(b)^b * P(c)^c}{\frac{A}{N} * \frac{B}{N} * \frac{C}{N}}$

分子をテキストにおける単語aの出現頻度を単語aの数で累乗していて、P(a)がa回起こる確率を示している。



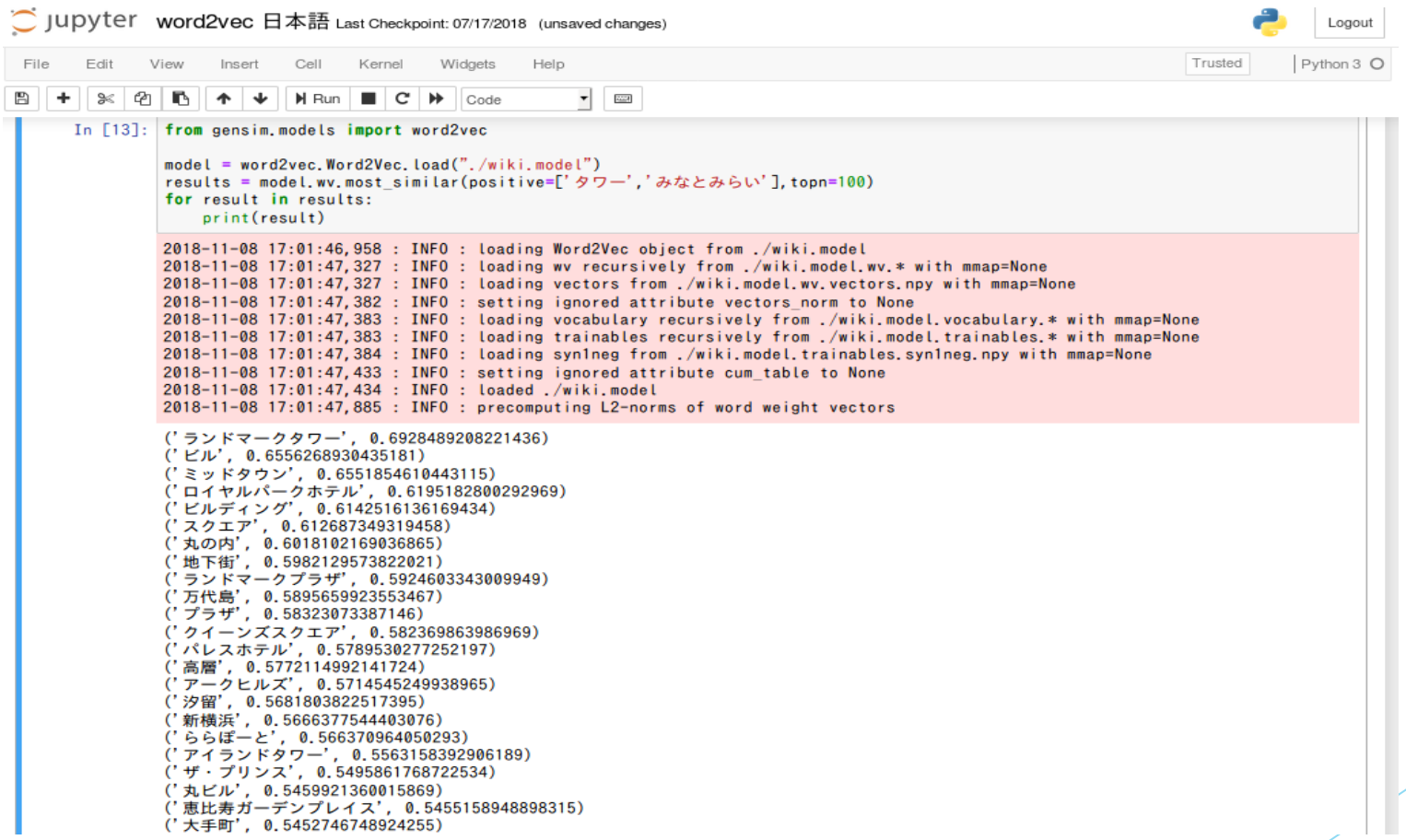
実験

- 使用言語:python3.5.2
- 使用環境:ubuntu16.04
- 使用テキスト:日本語wiki(3.3GB)

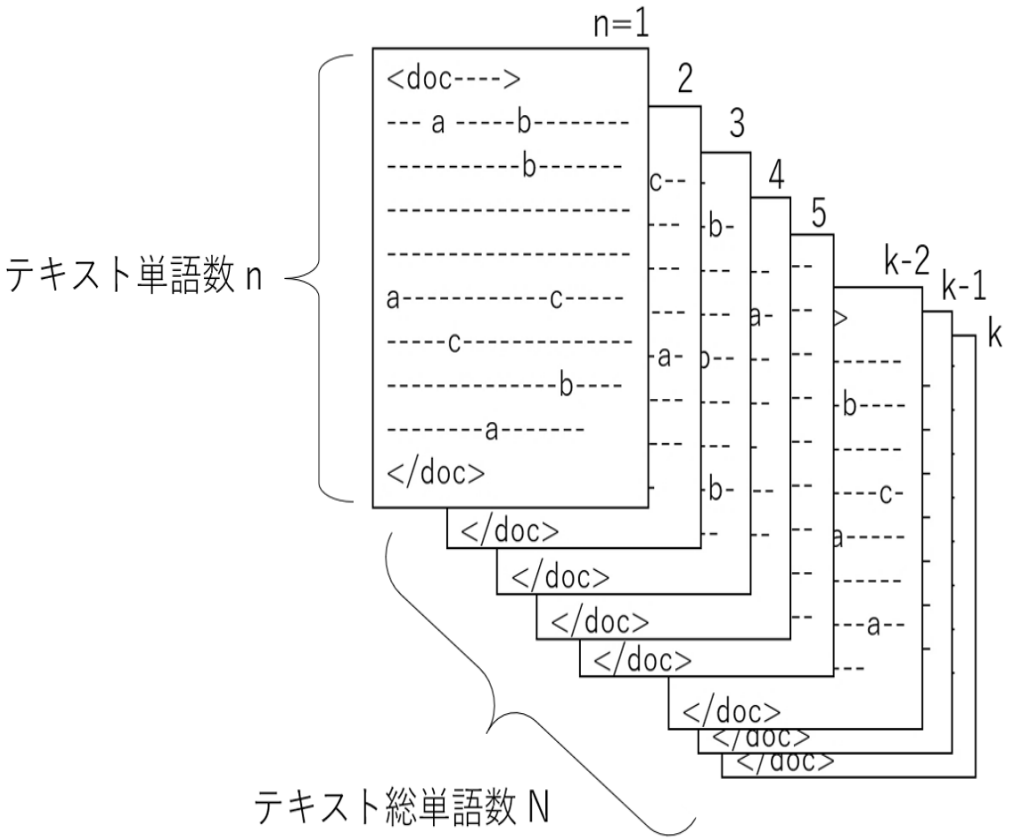
- 以下に実験で行った手順を記す。
- 1.大量のテキスト(日本語wiki3.3GB)をMeCabを用いて形態素解析し、Word2Vecでa、b→cという推論規則のPMIを求める。

実験

- Wiki(3.3Gb)のテキストからword2vecでみなとみらい、タワーを入力
- みなとみらい,タワー⇒ランドマークの推論を仮定する。



実験



wikiテキストのイメージ

2.wikiテキスト(3.3GB)の総単語数を求め、テキスト(3.3GB)に含む単語aの総数を求める。
単語b及びcも同様に行う。

3.a、b、cという単語を全てを含むテキストを抽出し、<doc～>-</doc>で括られたテキストごとにそれぞれの単語数を求める。

実験

- 4.PMMIの式に代入し、<doc～>-</doc>のテキストごとにPMMIを求める。
- 5.テキストを読み、自身の主観でテキストを4段階評価する。
- 6.PMMIと自身の主観で4段階評価したものをテキストごとに比較する。

$$\mathrm{PMMI}(x, y, z) = \log_2 \frac{p(x,y,z)}{p(x)p(y)p(z)}$$

$$= \log_2 \frac{p(x,y,z)}{p(x)p(y)p(Z)}$$

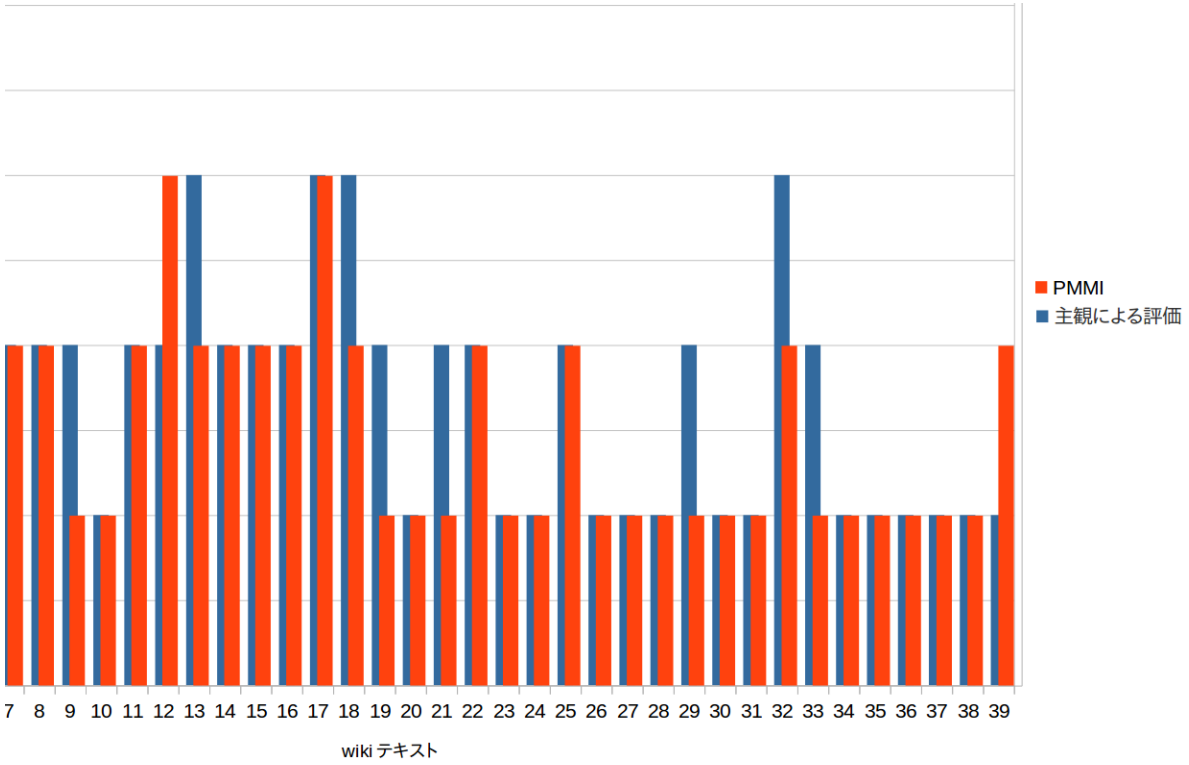
結果

テキストのタイトル	みなとみらい	タワー	ランドマーク	単語数	PMMI	主観評価
高島(横浜市)	10	3	0	2248	37.3713134	2
都市景観100選	1	1	0	2260	-0.6258973	1
ブルーダル	1	1	1	361	7.31285879	2
横浜ランドマークタワー	8	23	20	4367	211.00363	4
横浜市営バス	8	1	0	19051	14.1476866	2
神奈川区	1	4	0	3185	5.8891909	1
新高島	10	1	0	1611	34.0584825	2
馬車道	8	1	0	1356	25.5849995	2
横浜市営バス浅間町営業所	1	3	0	6467	-0.42133131	1
特別展マンモスYUKA	2	1	1	2810	0.43136063	1
Audiみなとみらい	6	1	1	790	19.4330723	2
栄本町線	14	1	0	2655	51.9799043	3
みなとみらいグランドセントラルタワー	9	2	0	1135	32.8843191	2
横浜スカイウォーク	1	6	0	1574	16.4495394	2
みなとみらいセンタービル	7	1	0	549	25.1499213	2
北仲通地区	4	5	0	4460	16.0415804	2
グランモール公園	17	7	3	4006	90.789824	3
コットンハーバー地区	5	3	0	3812	13.4759546	2
神奈川県立県民ホール	1	1	0	2423	-0.9273122	1
都心	2	1	0	11779	-5.77136446	1
横浜紅葉坂	1	1	1	261	8.71670588	1
横浜市	8	3	2	15739	21.7291426	2
イオン本牧	2	1	1	2103	1.68572447	1
若葉台(稲城市)	1	1	1	771	4.02866272	1
横浜幻想	3	1	1	572	10.0756974	2
東海道本線	1	4	0	27517	-3.44368284	1
渋谷駅	1	1	0	12699	-8.09685795	1
岩倉榮利	1	1	0	711	4.37930662	1
内田町(横浜市)	3	1	1	673	9.37192329	1
金沢地先埋立事業	2	1	0	7325	-3.71541526	1
KARA	1	1	0	10484	-7.26728104	1
横浜メディアタワー	2	4	1	493	15.9640924	2
アメリカ山公園	1	2	1	896	5.37835911	1

PMMI
- 100以上
- 30-100
- 10-30
- 10未満

主観評価
- よく分かる
- まあまあ分かる
- あまり分からない
- 全く分からない

結果



PMMIと主観の比較

- ４段階評価したものをグラフに表したため評価が１のテキストが多く、関連性が高く見える。
- PMMIが大きいということが、その単語が含まれるテキストの文脈の類似と関係が深いことを分析した。
- 文を読まなくてもPMMIを求めることで文脈を判断する指標になる。

今後の課題

- しかし今回は私一人の主観評価による比較だったので、多数の主観評価をする必要があった。
- また本研究では3つの単語におけるPMMIを求めたが、4つ以上の単語であってもPMMIを求めることができると考えられる。
- AIによるPMMIの自動評価も考えられる。

Word2Vecとは

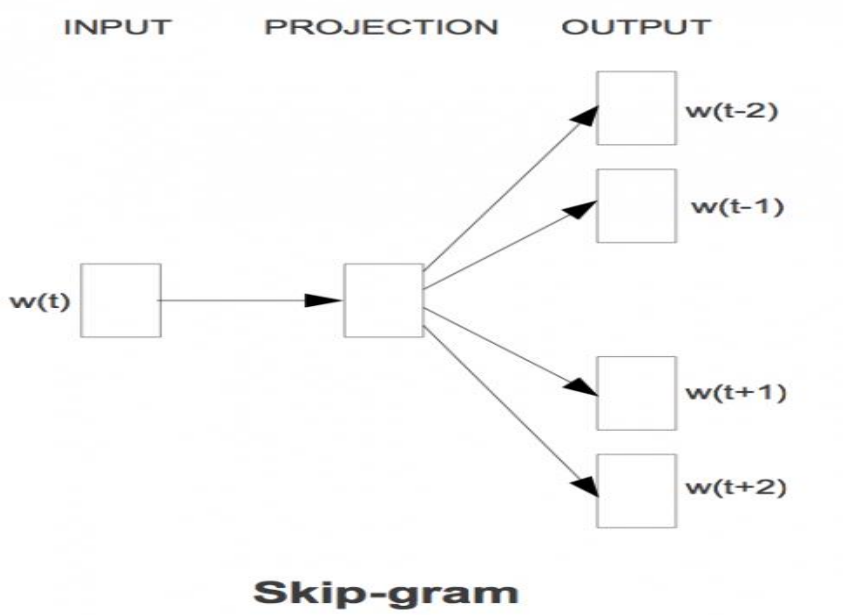
- 大量のテキストデータを解析し、各単語をベクトル化する手法
- 先行研究よりWord2Vecの類似度の計算は、
  単語間のPMIを求めるのと等価であると示されている

PMI

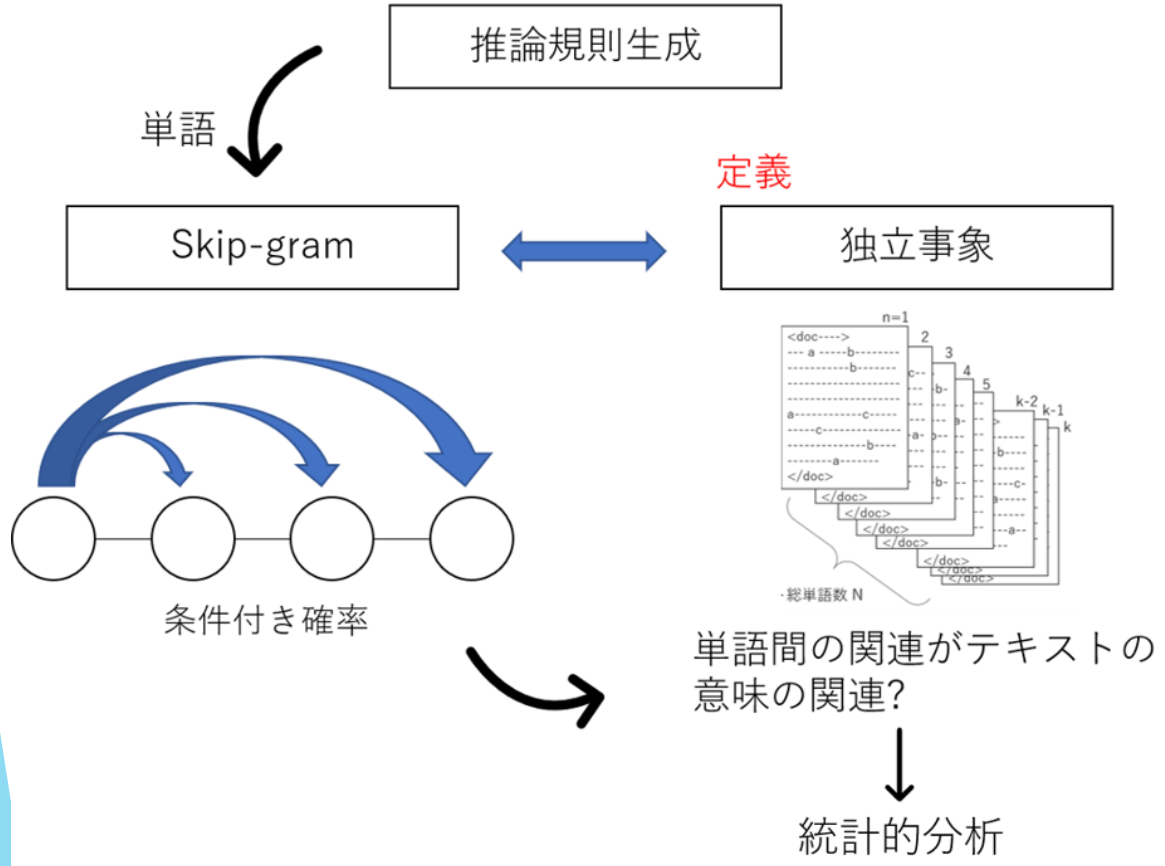
- 確率変数のある実現値xと, 別の確率変数のある実現値yに対して, 自己相互情報量PMI(x, y)は,
- $\mathrm{PMI}(x, y) = \log_2 \frac{p(x,y)}{p(x)p(y)}$
- と定義され, 値が大きければ大きいほどxとyの関連している度合いが強い.

Skip－Gram　Model

- Skip-gram は 着目してる中心の単語からその文脈を構成する単語を推定
- Skip-gram モデルで得られた単語 ベクトルを利用すると、単語の意味の類似度が計算できる



目的



先行研究では推論攻撃に関する研究が行われているが、いずれの研究でも推論規則の効率的な獲得は行われていない

↓

Word2Vecを利用することが効率的な推論規則生成では？

Aという文字がa回起きたことを意味するp(a)^aをすることが必要だと考えた

Skip-Gramによる確率では