

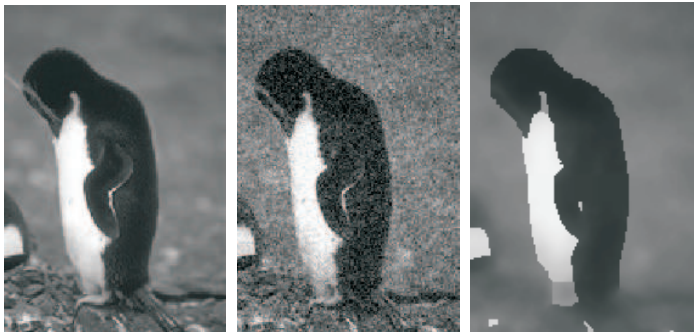
Примеры решения практических задач при помощи графических моделей

«Графические модели»

Данные со структурой

- Изображения
- Видео
- Сигналы, звук
- 3D облака точек
- Тексты
- Задачи о размещениях
- Коды
- Молекулярная биология

Восстановление изображений



$$E(T) = \sum_{i \in \mathcal{V}} E_i(t_i, l_i) + \alpha \sum_{(i,j) \in \mathcal{E}} E_{ij}(t_i, t_j), \quad t_i = 0, \dots, 255$$

t_i – цвета очищенного изображения

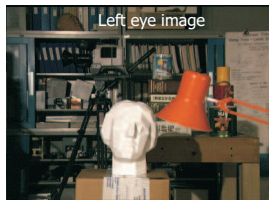
Сегментация изображений



$$E(T) = \sum_{i \in \mathcal{V}} E_i(t_i) + \alpha \sum_{(i,j) \in \mathcal{E}} E_{ij}(t_i, t_j), \quad t_i = 1, \dots, K$$

t_i – метки классов пикселей

Стерео



$$E(T) = \sum_{i \in \mathcal{V}} E_i(t_i) + \alpha \sum_{(i,j) \in \mathcal{E}} E_{ij}(t_i, t_j), \quad t_i = -D, \dots, 0, \dots, D$$

t_i – сдвиг пикселя до соответствия (диспаритет)

Склейка изображений



$$E(T) = \sum_{i \in \mathcal{V}} E_i(t_i) + \alpha \sum_{(i,j) \in \mathcal{E}} E_{ij}(t_i, t_j), \quad t_i = 1, \dots, K$$

t_i – номер изображения, из которого берется пиксель i

Панорамы: потенциалы

Унарные потенциалы не 0 только для семян:

$$E_i(t_i) = \begin{cases} +\infty, & i \in \text{Seed}_k, \quad k \neq t_i, \\ 0, & \text{иначе.} \end{cases}$$

Бинарные потенциалы:

- $E_{ij}(t_i, t_j) = |I_i(t_i) - I_i(t_j)| + |I_j(t_i) - I_j(t_j)|,$
- $E_{ij}(t_i, t_j) = \frac{|I_i(t_i) - I_i(t_j)| + |I_j(t_i) - I_j(t_j)|}{|\nabla_{ij} I(t_i)| + |\nabla_{ij} I(t_j)|}$

Поиск составных объектов на изображении



$$E(T) = \sum_{i \in \mathcal{V}} E_i(t_i) + \alpha \sum_{(i,j) \in \mathcal{E}} E_{ij}(t_i, t_j), \quad t_i = 1, \dots, K$$

t_i – позиция части номер i на изображении

Поиск объектов: потенциалы

Модель лица:



Параметры каждой части – позиция.

Унарные потенциалы:

$$E_i(\mathbf{t}_i) = -\log \mathcal{N}(\beta(\mathbf{t}_i) | \mu_i, \Sigma_i)$$

$\beta(\mathbf{t}_i)$ – линейные фильтры, примененные в точке \mathbf{t}_i

Бинарные потенциалы:

$$E_{ij}(\mathbf{t}_i, \mathbf{t}_j) = -\log \mathcal{N}(\mathbf{t}_i - \mathbf{t}_j | \mu_{ij}, \Sigma_{ij})$$

Видео: сегментация объекта



Видео: оптические потоки

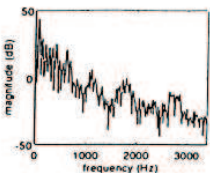
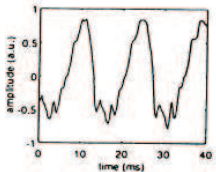
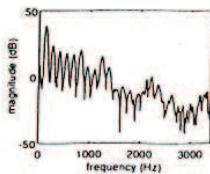
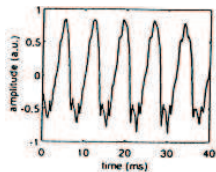
Переменные t_i – смещение пикселя относительно предыдущего кадра.

$$E(T) = \sum_{i \in \mathcal{V}} E_i(t_i) + \alpha \sum_{(i,j) \in \mathcal{E}} E_{ij}(t_i, t_j), \quad t_i = 1, \dots, K,$$

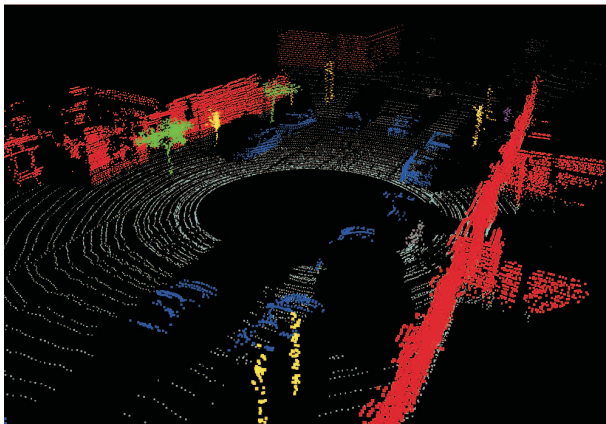
$$E_i(t_i) = d(I_i, I_{i+t_i})$$

$$E_{ij}(t_i, t_j) = \|t_i - t_j\|^2.$$

Сегментация сигналов: звук



Трёхмерные облака точек



Раскраска карты



$$E(T) = \sum_i E_i(t_i) + \alpha \sum_{(i,j) \in \mathcal{E}} E_{ij}(t_i, t_j), \quad t_i - \text{цвет страны}$$

- $E_i(t_i)$ = предпочтения цвета
- $E_{ij}(k, \ell) = M[k = \ell]$

Размещение названий на карте



$$E(T) = \sum_i E_i(t_i) + \alpha \sum_{(i,j) \in \mathcal{E}} E_{ij}(t_i, t_j), \quad t_i \text{ — позиция надписи}$$

- $t_i \in \{\emptyset, \leftarrow, \rightarrow, \uparrow, \downarrow, \nearrow, \searrow, \swarrow, \nwarrow\}$
- $E_i(y_i) =$ штраф, если надписи нет
- $E_{ij}(k, \ell) = \infty,$ если i -я надпись в позиции k перекрывает j -ю в позиции ℓ

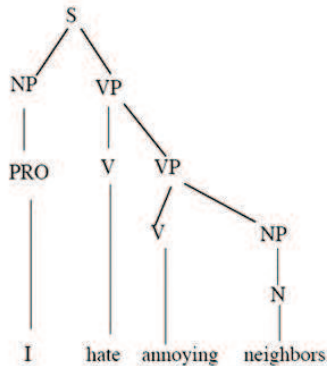
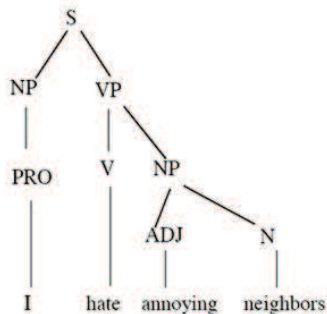
Тексты: задачи

- Сегментация слов на части речи (POS-tagging)
- Распознавание имен собственных (named entity recognition)
- Выравнивание слов для машинного перевода
- Сегментация китайского текста на слова
- Разбор предложений при помощи контекстно-свободной грамматики

Сегментация частей речи: Trigrams'n'Tags (TnT)

- Скрытая марковская модель (НММ) второго порядка
- Наблюдаемые переменные – слова
- Скрытые переменные – метки частей речи
- $P(t_i | t_{i-1}, t_{i-2})$ оцениваются по размеченному корпусу текстов
- Сглаживание вероятностей, обработка новых слов, и др.

Сегментация частей речи: грамматики



Блочное кодирование для передачи данных по каналу с помехами

$$z_1 \dots z_k \xrightarrow{\text{кодирование}} y_1 \dots y_n \xrightarrow{\text{канал с шумом}} \bar{y}_1 \dots \bar{y}_n \xrightarrow{\text{декодирование}} \hat{z}_1 \dots \hat{z}_k$$

Скорость передачи — $\frac{k}{n}$.

Пример: код Хэмминга ($k = 4$, $n = 7$)

$$y_1 = z_1, y_2 = z_2, y_3 = z_3, y_4 = z_4,$$

$$y_5 = z_1 + z_2 + z_3,$$

$$y_6 = z_2 + z_3 + z_4,$$

$$y_7 = z_1 + z_3 + z_4.$$

Теория кодирования

Теорема Шеннона (1947): для любого канала существует **пропускная способность**, такая что

- Для любой скорости передачи сообщений меньше **пропускной способности** существуют **идеальные** коды, вероятность ошибки декодирования которых стремится к нулю при стремлении длины блока к бесконечности.
- Для любой скорости передачи сообщений больше **пропускной способности** **идеальных** кодов не существует.

Пример **идеального** кода: каждому **\mathbf{z}** сопоставить случайный **\mathbf{y}** в $1/(\text{пропускная способность})$ раз длиннее.

Проблема: не известно **идеальных** кодов с полиномиальным временем декодирования.

Линейные коды

Линейный код задается $\mathbf{y} = \mathbf{zG}$. \mathbf{G} – генераторная матрица.
Для кода Хэмминга:

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

Также код можно задавать проверочной матрицей \mathbf{H} : $\mathbf{Hy}^T = \mathbf{0}$.

$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Графическая модель

$$\begin{aligned} P(\mathbf{y} | \bar{\mathbf{y}}) &\sim \underbrace{[\exists \mathbf{z} : \mathbf{y} = C(\mathbf{z})]}_{\text{кодовые слова}} \underbrace{P(\bar{\mathbf{y}} | \mathbf{y})}_{\text{шум в канале}} = \\ &= \prod_{\ell} \left(1 + \sum_{t: H_{\ell t} = 1} y_t \right) \prod_i P(\bar{y}_i | y_i) \end{aligned}$$

Получается графическая модель с большим количеством факторов высоких порядков.

Идея (Gallager, 1960, MacKay&Neal, 1996): низкоплотный код (LDPC-код)

- H – разреженная матрица.
- Подсчитать маргинальные распределения через циклическую передачу сообщений.