

# RPAによる窓口業務シフト 作成の自動化



## Code for Bunkyo

荒川清晟<sup>1,2</sup>, 野寄修平<sup>1,2</sup>, 下笠博敏<sup>3</sup>, 梅田裕次<sup>3</sup>

1. 東京大学GCL, 2. Global Creative Leaders Society,  
3. 東京都文京区

# 文京区の課題の抽出 (2018年度)

- 文京区民, 学生を対象に「**区の公共サービスに期待すること**」をテーマにワークショップを開催
  - ワークショップでの話題:  
子育て・教育・コミュニティ・介護・行政手続き



# 文京区の課題の抽出 (2018年度)

- 文京区民, 学生を対象に「**区の公共サービスに期待すること**」をテーマにワークショップを開催
  - ワークショップでの話題:  
**子育て**・教育・コミュニティ・介護・**行政手続き**

## 子育てワンストップサービス

(COG2018, Code for Bunkyo)



LINEチャットボットを利用した緊急一時保育の利用予約サービス

## 窓口シフト自動作成システム

(2018年からの取り組み)



シフト作成システム

ロボティック・プロセス・オートメーション (RPA)によりシフト作成を自動化

# アイデアの背景

- 窓口業務が主である文京区戸籍住民課住民記録係において、窓口対応のシフト作成という、本来の業務に付随して生じる作業により本来の業務を遂行する時間が削られてしまっている
- 総務省主催の「地方自治体における業務プロセス・システムの標準化及びAI・ロボティクスの活用に関する研究会（以下スマート自治体研究会）」では、「住民・企業等にとっての利便性向上」・「自治体の人的・財政的負担の軽減」を目的として、システムやAI（人工知能）等の技術を駆使して効果的・効率的に行政サービスを提供する自治体を「スマート自治体」とし、ベストプラクティスの横展開を図ろうとしている

# アイデアの概要

## ■シフト作成の条件

- ・係長のいずれかが1名入る
- ・主査のいずれかが1名入る
- ・チームのいずれかが1名入る
- ・支援のいずれかが1名入る

すごくたくさんある

## ■職員の休み希望

## ■休日（土・日・祝）

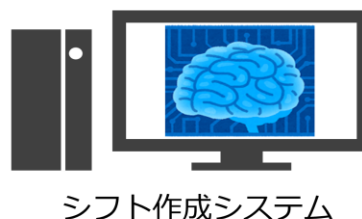
これまで



## ■シフト表を作成

日				1	2
月	3	4	5	6	7
火					
水					
木					
金					
土					
日					
祝					

これから



毎月2人日かかる作業が  
RPAにより数分で終了

シフト作成担当の職員は  
窓口業務に時間を使える

シフト作成熟練の職員が行っていたシフト作成をコンピュータが代行することで、職員が本来の業務を行なう時間を増やす

# 区民サービスの質・量がさらに向上!!

# RPAによるシフト作成自動化 (1/2)

## ① 自然言語から数式への変換

自然言語（日本語）で記述されたシフトの条件

満たす必要があるもの

「必ず」という言葉が含まれている条件、人間が指定した条件

制約条件とする

例) 同じ人は同じ日の違うシフトに入れない

$$\forall i, j, x_{ij1} + x_{ij2} + x_{ij3} \leq 1$$

優先度が低いもの

「できれば」「なるべく」といった言葉のある条件

目的関数に含める

条件を満たさないと目的関数 $f(X)$ が大きくなる

目的関数  $f(X)$ ,  $X = [x_{000}, x_{001}, \dots, x_{lmn}]$   
 $x_{ijk}$  ( $i$ : 人のID,  $j$ : 日付, シフトの種類) =  $\{0, 1\}$

# RPAによるシフト作成自動化 (2/2)

## ②数式を最適化問題として解く 制約条件の中で目的関数 $f(X)$ を 最小にする $X$ を探索する

```
# 朝, レジ前日と夜同時禁止条件, 夜→朝禁止条件
for d in wednesday_indexes:
    for m, member in members.iterrows():
        # 足したら1以下 ⇒ 両方11を禁止
        mymodel.add_constraint(x[d, m, inv_shifts["朝"]] + x[d, m, inv_shifts["夜"]] <= 1)
        mymodel.add_constraint(x[d, m, inv_shifts["レジ前日"]] + x[d, m, inv_shifts["夜"]] <= 1)
for d in thursday_indexes:
    if d > 0 and days[d-1] == days[d] - ONE_DAY:
        for m, member in members.iterrows():
            mymodel.add_constraint(x[d - 1, m, inv_shifts["夜"]] + x[d, m, inv_shifts["朝"]] <= 1)
```

条件を満たすシフト表を自動で出力

# スケジュール

