An Approach to AI based Assistance for Asylum Applications in Germany using the LICON Framework

The German asylum policy is alone the world, providing a subjective right for political refugees to be granted asylum. The right of asylum is not the right of the state to grant asylum, to be held against the persecuting state but the right of the persecuted individual, to be held against the receiving state. This implies the unique challenge of validating every asylum application on its own, with every case being unique. Due to the increasing number of refugees in the world, carrying out this validation, whether asylum is granted or not, became more and more difficult. Every validation takes months, and in combination with too little workforce and high costs, this is a topic where improvement is necessary. The main goal behind this research proposal is not to automate the process of decision making whether to grant someone asylum or not with Artificial Intelligence. Instead, it aims to use Artificial Intelligence to provide auxiliary help for the people making these decisions in speeding the process up and to making it fair in a way that minimizes and the human bias that is part of every action an induvial human takes

Artificial Intelligence has emerged as one of the trending topics regarding Data Analytics in the last few years. With Big Data and its increasing capabilities, several new opportunities, and possible fields of application emerged. To apply it in the scenario of asylum decision transparency is a key point, as it is for any technical solution that approaches this problem. If people are not granted asylum, it is their right to know about the factors that led to this decision, and this limits the technical solutions regarding complexity.To provide transparency, it is crucial to implement the neural network in a way that the influence of each input value can be quantified with a precise method to determine which impact it has on the overall output value. The Local Input COntribution in Neural Networks (LICON), a generic framework that was published by Gjergji Kasneci and Thomas Gottron, approaches exactly this issue.It uses a linear approximation of a function F in a local neighborhood around the input values Xi at each neuron of the neural network. This leads to the overall error of the neural network depending solely on the errors made in the estimation of the behavior of single neurons. This is used in a gradient approach, which serves an iterative method to calculate local linear approximations for each neuron. This approach makes it possible to measure the influence of each input variable on the change of the neuron output value.

.

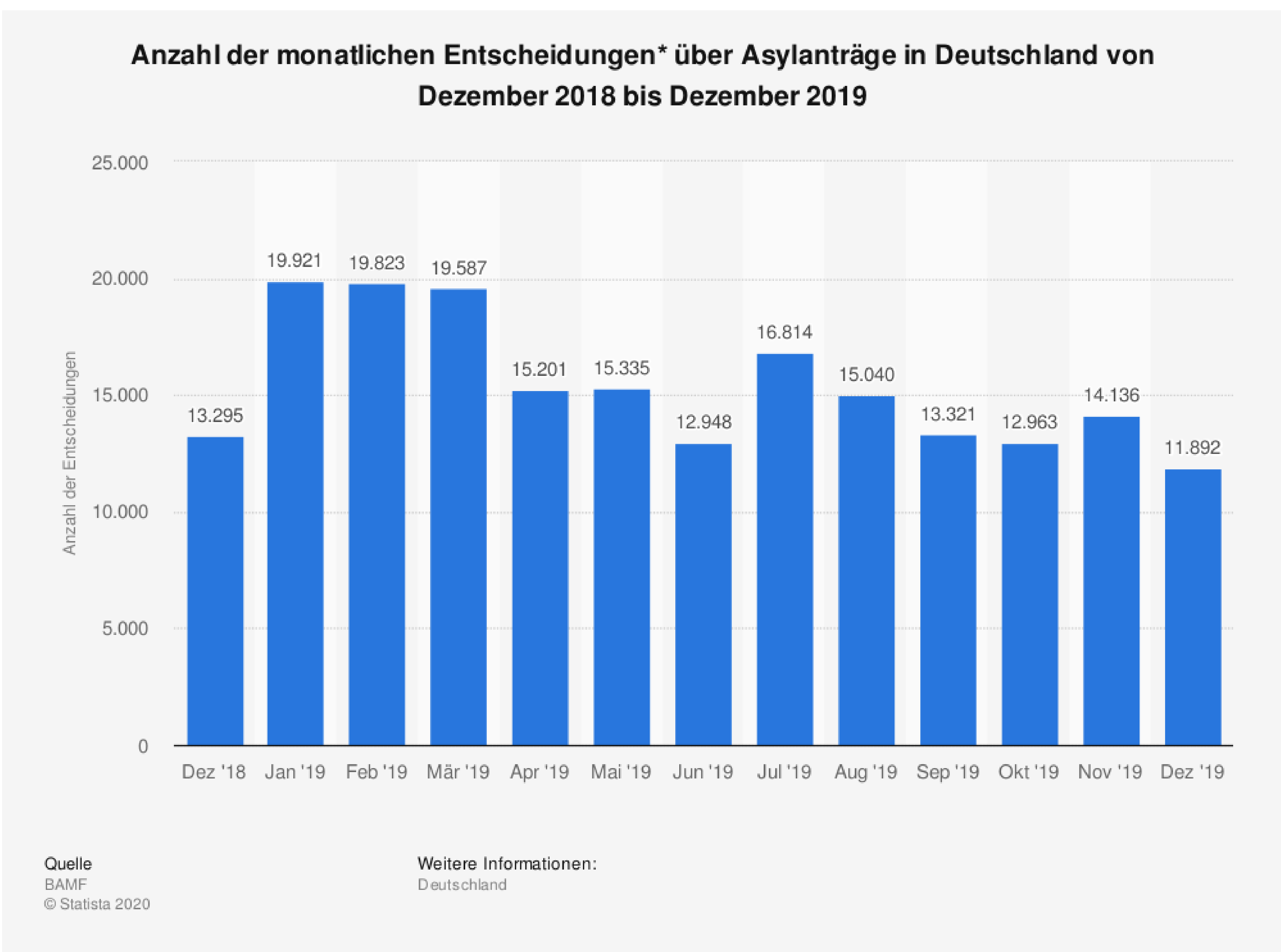

Figure 1. Number of asylum decision per month

Algorithm 1 LocalContribution(DANN with nodes $z_i^{(l)}$, input values $\mathbf{x}_i^{(0)}$)

1: $\alpha_{ji}^{(0)} \leftarrow \Delta_{ji} \forall j, i = 1, \ldots, n_0$
{Kronecker Δ_{ji} initialises each input variable to only influence itself; that is, each $\alpha_{ii}^{(0)}$ is set to 1 and for all $j \neq i$ to 0.}
2: **for** $l = 1 \ldots m$ **do**
3: **for** $i = 1 \ldots n_l$ **do**
4: $\vec{\beta}_i^{(l)} \leftarrow \vec{\nabla}\left(f_i^{(l)}\left(\sum_{j \in \mathrm{IN}(i)} w_{ji}^{(l)} \cdot \mathbf{x}_j^{(l-1)}\right)\right)$
{Compute gradient to model local influence of the input values to a neuron.}
5: **for** $k = 1 \ldots n_0$ **do**
6: $\alpha_{ki}^{(l)} = \left(\sum_{j \in \mathrm{IN}(i)} \beta_{ji}^{(l)} \cdot \alpha_{kj}^{(l-1)}\right)$
{Aggregation into linear model for current layer.}
7: **end for**
8: **end for**
9: **end for**
10: **return** $\vec{\alpha}^{(m)}$

Figure 2. LICON Meta-Algortihm

Jonas Bücker
Master Cybersecurity

Lecture Series on Practical Data Engineering
Hasso Plattner Institute, Potsdam, Germany

E-Mail: jonas.buecker@students.hpi.de

HPI Hasso Plattner Institut