UDC 519.6:94
https://doi.org/10.31891/csit-2022-4-9

OLEKSANDR MELNYKOV, ANDREI KAPELESHCHUK
Donbas State Engineering Academy, Kramatorsk, Ukraine

WEB-BASED SYSTEM of DECISION SUPPORT FOR CALCULATING COMBAT AND NON-COMBAT LOSSES DURING MILITARY CAMPAIGNS IN THE MIDDLE AGES

The project is devoted to calculating the number of losses of medieval armies using artificial intelligence methods. Several possible calculation methods have been considered, and historical sources have been analyzed. The analysis of factors that affected combat and non-combat losses and the analysis of scientific and popular science literature have been carried out, and the methods proposed by specialists have been translated into the language of formulas. The process of building neural networks, selecting their architecture, searching and synthesizing data for training samples have been considered, and the process of training and verifying the obtained data have been considered. Two models of neural networks and an information model in the form of UML diagrams of the future web application have been developed. Diagrams of use cases, classes, and components for each element of the web system have been described. The models have been implemented using the modern Django framework. A full-fledged web application with microblogging has been developed and tested. The conclusions have been drawn about the efficiency and scalability of the developed system, and the functionality of the system has been demonstrated using a real historical example.

The project has developed the models of an information system for analyzing the number of troops and calculating military losses of the remote past. Mathematical models have been described, historical sources have been analyzed, and detailed models of the application have been created using the UML modeling language, which allows you to understand its interface in detail at the modeling stage.

Based on the data obtained in the analysis of historical literature, the neural network architectures have been developed to determine non-combat losses in the medieval army and determine combat losses based on the data on the number of each branch's soldiers of the armed forces who were involved during the battle.

Web applications for calculating combat and non-combat losses have been developed, interface design and mini-block for publishing system News have been developed either. The non-combat losses suffered by the Mongol army in the first month of the Western campaign (1236) have been calculated as the example.

Keywords: medieval military history, calculating the number of troops, Perceptron, modeling, Python, Django.

ОЛЕКСАНДР МЕЛЬНИКОВ, АНДРІЙ КАПЕЛЕЩУК
Донбаська державна машинобудівна академія

WEB-СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ДЛЯ ПІДРАХУНКУ БОЙОВИХ ТА НЕ БОЙОВИХ ВТРАТ ПІД ЧАС ВОЄННИХ ПОХОДІВ У СЕРЕДНЬОВІЧЧЯ

Робота присвячена розрахунку чисельності втрат середньовічних армій за допомогою методів штучного інтелекту. Розглянуто декілька можливих методів підрахунку, проаналізовано історичні джерела. Проведено аналіз факторів, що впливали на бойові та не бойові втрати, аналіз наукової та науково-популярної літератури та перекладено запропоновані спеціалістами методи на мову формул. Розглянуто процес побудови нейронних мереж, підбору їх архітектури, пошуку та синтезу даних для навчальних вибірок, розглянуто процес навчання та перевірки отриманих даних. Розроблено дві моделі нейронних мереж та інформаційна модель у вигляді UML-діаграм майбутнього веб-додатку. Описано діаграми варіантів використання, класів та компонентів для кожного елементу web-системи. Виконано реалізацію моделей за допомогою сучасного фреймворку Django. Розроблено та опротестовано повноцінний web-додаток з мікроблогом. Зроблено висновки щодо ефективності та можливості масштабування розробленої системи, продемонстровано функціонал системи на реальному історичному прикладі.

У рамках проекту розроблено моделі інформаційної системи для аналізу чисельності та підрахунку військових втрат далекого минулого. Описано математичні моделі, проаналізовано історичні джерела та створено детальні моделі додатку за допомогою мови моделювання UML, що дозволяє детально зрозуміти його інтерфейс на етапі моделювання.

На основі даних, отриманих під час аналізу історичної літератури, розроблено архітектури нейронної мережі для визначення небойових втрат середньовічної армії та визначення бойових втрат на основі даних про кількість солдатів кожного з видів збройних сил, які були загиблі. брав участь під час бою.

Розроблено веб-додатки для розрахунку бойових і небойових втрат, розроблено дизайн інтерфейсу та міні-блок для публікації системи Новини. Для прикладу підраховано небойові втрати монгольської армії в перший місяць західного походу (1236 р.).

Ключові слова: середньовічна військова історія, розрахунок чисельності військ, персептрон, моделювання, Python, Dgango.

Introduction

War is a phenomenon that accompanies humanity in all historical time, so the question of the number of troops and military losses was also raised by Herodotus. Unfortunately, there is currently no single system for calculating combat losses, and even losses in fairly modern conflicts are mostly estimated approximately, and during the constant wars of the Middle Ages, there was simply no one to calculate the military losses. It should be noted that by the end of the XIX century, up to 70% of losses fell on the so-called non-combat losses (death from diseases, injuries, cold, etc.).

When it comes to estimating losses, first you need to estimate the size of armies, because the number of troops directly affects losses, but the number of armies directly depends on losses. During the three centuries of the development of modern historical science, historians have developed several approaches to estimating the size of the medieval army:

The first and oldest approach was to directly reading historical sources and trusting them. But in the chronicles, either there is no information about the number of troops at all, or their number was determined in an unknown way (it is not clear who counted the soldiers, considering that the chronicles, as a rule, were created several decades after the events described in them). The Chronicles rather show the chronicler's attitude to certain events: for example, in the European medieval chronicles, there were always thousands of times more dead pagans than Christians, and it may seem that the Invincible Knight's army came out of the battle without any losses, but this is not like that.

The second approach is based on calculating the number of troops by determining the number of soldiers in one combat unit, and then determining the number of these combat units. Many historians use this method, but it does not take into account the division into combat and non-combat losses and assumes that each "thousand" has a thousand soldiers, and not less or more soldiers. But even in any modern army, there is no fully equipped military unit (even in those that do not fight), not to mention the colossal non-combat losses during the marches.

Another approach involves calculating the number of troops, which is based on the total number of inhabitants of the country and on the size of the enemy army (of course, no one takes more troops than it is necessary). Based on the archaeological data (the area of cities, the number of towns, etc.), knowing the amount of financial resources needed for the birth of one warrior (especially it is typical for the Knight armies), you can calculate the mobilization reserve of the country, and relying on it you can calculate the number of troops that a particular country could send for perform combat tasks. An important role in assessing the mobilization reserve is played by the type of army that the country used. Nowadays, the mobilization reserve is almost all males of military age, and some military affairs can be taught for a few months. The model of the knight's army assumes the presence of a professional army only (chivalrous military affairs were studied for years), and the Knight's equipment required very large financial costs.

The calculated number of troops can be checked by using additional calculations of the amount of food for feeding the army, the length of the wagon train, and other factors, checking the calculated values for adequacy. For example, for each mounted warrior at different times there were two or three horses, and horses or oxen were also needed to transport wagons. According to the standards, with an average load, a medium-sized horse needs 13 feed units, that is 8 kg of high-quality hay. To maintain the mobility of the army, it is necessary to keep a reserve for several days with you, especially during winter hikes, when it is not possible to graze animals. It should be noted that during the summer period, there was also no great opportunity for grazing animals during crossings, because the grazing process takes a long time, but during military campaigns, the main task was to walk as much distance as possible in a day [1]. No more than 70 kg of hay can be loaded on one cart, wood can no longer withstand, and one cart with a horse occupies at least 10 meters on the march, a horse-drawn guesser occupies at least 6 meters. Knowing this, it is not difficult to calculate the lengths of columns, the movement of which had to be controlled without any means of communication.

The main reasons for the extremely large non-combat losses in the XIX century were: epidemics, lack of medicine, lack of field medicine, lack of a centralized food supply system (each unit supplied its own food, feeding on the site), lack of any living conditions (soldiers slept on the ground, tents were not existed). On the other side, people in the XIII century could be physically stronger and more adapted to severe weather conditions (which is confirmed by some sources) than people of the XIX century, but somehow it is impossible to measure the average strength and compare it with previous measurements, this is a rather subjective parameter. To predict non-combat losses, you can trace the main parameters that determine human health: air temperature, precipitation, and so on. Also, some data on the diseases or nutrition of people of that time can be given to us by the archaeological sources or miniatures of that time.

Air temperature is one of the most important factors, knowing which, you can calculate the number of deaths from seasonal infections, pneumonia, dehydration or frostbite. To calculate this parameter, you need to know in which climatic zone events occurred, at what time of the year and approximate years (it is known that in 1200-1300 in Europe it was much warmer than in the XX century [2]). Another equally important parameter is the terrain through which the army passes, the probability of being injured on the march or becoming a victim of poisonous or predatory animals depends on the terrain.

In the complex "time of the year-terrain-climate zone" you can describe most of the phenomena that lead to non-combat losses, but for a more accurate calculation, you also need to take into account the number of days in the campaign, since most viruses have an incubation period, which means that in the first weeks of the campaign, the intensity of this type of disease is lower.

On fig. 1 it is possible to notice the existence of some dependence of non-combat losses on the year, so we can assume the existence of a dependence on the weather conditions. (Jumps in the yellow and blue lines are caused by the plague epidemic in 1710-1711 [3, p. 58-59] – this suggests that large epidemics should be considered separately).

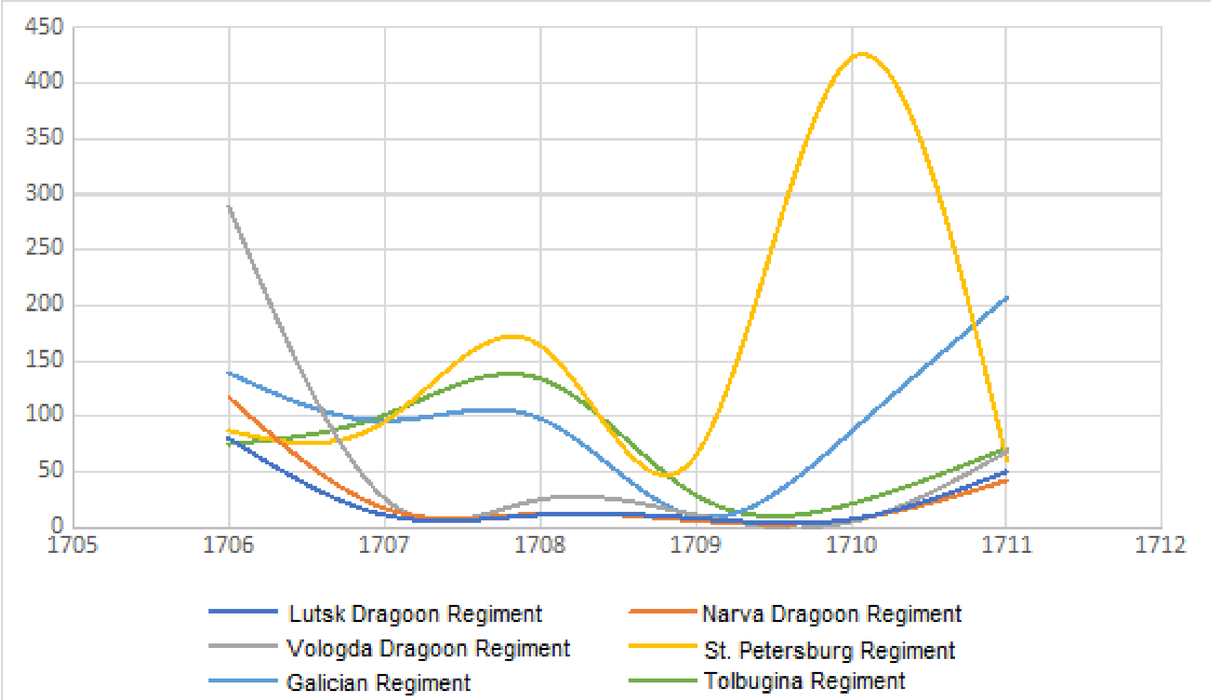

Fig. 1. Indicators of non-combat losses in various units of the Russian Empire from 1705 to 1711 [3]

Analysis of existing solutions

In 1916, the mathematician F. W. Lanchester developed a system consisted of two differential equations (1) and served to count the victims of wars:

$$\begin{cases} \frac{dx}{dt} = ax + bxy + cy + d \\ \frac{dy}{dt} = ey + fyx + gx + h \end{cases} \quad (1)$$

where:
a and e – speed of non-combat losses;
b and f – rate of losses due to impact on area targets;
c and g – losses from enemy actions on the front line;
d and h – incoming or outgoing reserves [6].

But this model is more adapted to the war of the late XIX - early XX century than for the wars of the Middle Ages, because such parameters as b and f are absent during medieval clashes, and primitive non-firearm artillery was used mainly for storming and breaking through the defenses of fortresses, rather than for destroying enemy manpower in an open field, for medieval battles the linear model of Osipov-Lanchester is more suitable.

Another approach to modeling considers combat in the form of a computer game with different levels of detailing, depending on the power of the computer (both an entire regiment and one warrior can be taken per unit, depending on the available computing resources). Most variants of such generalized models do not take into account the peculiarities of climate and terrain [7]. Some of the game projects of the Wargaming studio can be used to simulate the fighting of the XX century, due to the fact that they take into account many real physical indicators, modeling medieval battles with the help of games can provide some new information about the events. However, such modeling is very time-consuming, and most of the provided information will be superfluous in the task of simply counting warriors and losses.

Modeling

Let's create a mathematical model for calculating of non-combat losses. We have the following initial data:
t_{avg} – average monthly temperature;
T – deviation from the climatic optimum;
d – number of days on the hike;
k_t – terrain difficulty factor;
kol – number of warriors participating in the campaign;
os – average monthly precipitation.
a – non-combat losses per day.

The number of deaths from diseases can be calculated using formulas 3 and 4.

$$t = t_{cp} + T \tag{2}$$

$$c_1 = \begin{cases} \text{const1, at } t < 5 \\ \text{const2, at } 5 < t < 15 \\ \text{const3, at } t > 15 \end{cases} \tag{3}$$

$$d = consv + c_1 \tag{4}$$

Constants are the number of deaths as a percentage, determined by selection, based on the data from the wars of the XIX century. Constants 1-3 correspond to various climatic diseases (seasonal flu, hypothermia, cold, pneumonia, etc.), but constant v is responsible for diseases that do not depend on the season of the year.

Due to the fact that in most regions of Europe the amount of precipitation is quite small in summer, the coefficient is the lowest in summer, and in the winter months a good layer of snow, on the one hand, makes it difficult to move, on the other hand – places like swamps become less dangerous. The most dangerous months are March, April, October and November, when there is quite a lot of cold precipitation and the air temperature decreases. You can see that the "danger" increases in proportion to how much the air temperature decreases, and the amount of precipitation increases. For the winter months, the" danger " should be reduced. Let's write all mentioned above as Formula (5):

$$p = \frac{1}{t} * os - zile \tag{5}$$

where zile – the coefficient of cold winters and hot climate – it takes the value 0 at a comfortable plus temperature; at subzero or extremely plus temperatures – the value greater than 0.
Total non-combat losses per day:

$$a = (p + d)\frac{kol}{100} \tag{6}$$

Formula (6) does not take into account large epidemics, but it takes into account the usual set of viral and bacterial diseases. To calculate epidemics, we should add the percentage of deaths during the epidemic to the numbers *p* and *d*.

There is only the one uncertainty left in our model, namely the value of all our constants. They must be determined separately, calculating the result each time. Therefore, it is appropriate to use a direct propagation neural network – a perceptron.

At the perceptron inputs, we provide the following parameters: average temperature per month, average precipitation per month, terrain number, and the number of days on a hike. So on the first layer, we have got 4 neurons. We have 6 constants, the values of which need to be selected, so on the second layer we place 6 neurons, on the third layer – one output neuron, which gives out the percentage of non-combat losses per day. The neural network model is shown in fig. 2. Perceptron gives out the percentage of deaths per day.

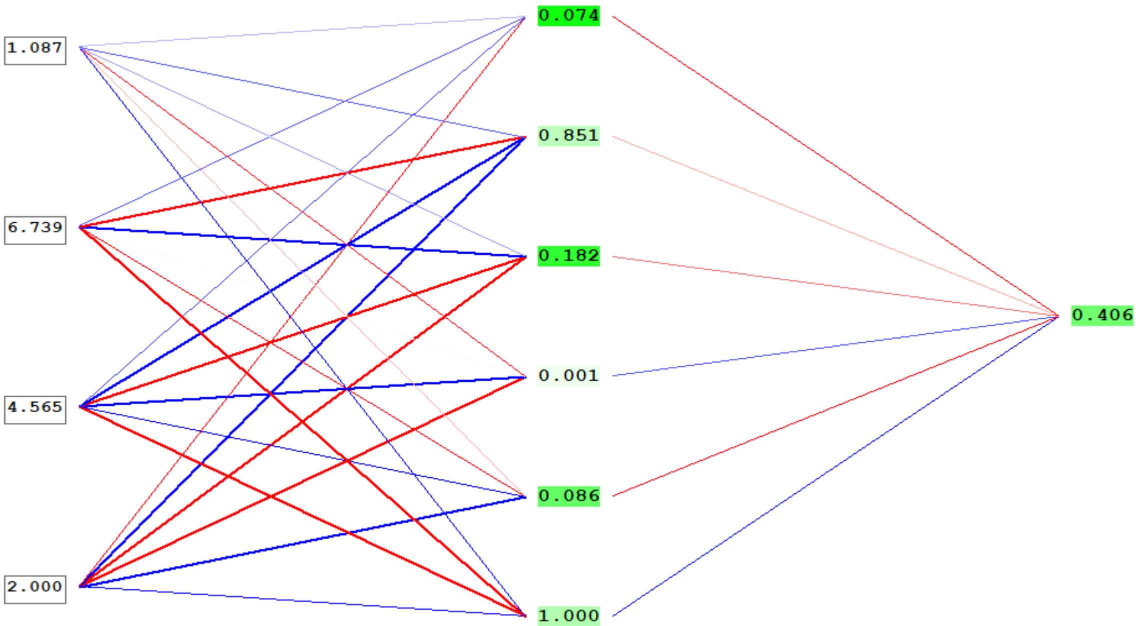

Fig. 2. Neural network model in the NeuroNet environment

But we take the average daily temperature and precipitation from fairly unverified sources (it is impossible to find out the amount of precipitation in the Middle Ages), so we propose a different neural network model that takes into account only the type of terrain, the time of the year, and the number of days in the hike (Fig. 3). Instead of temperature and precipitation data, we add two inner layers (which emulate temperature and precipitation) with 12 neurons, according to the number of months in the year.

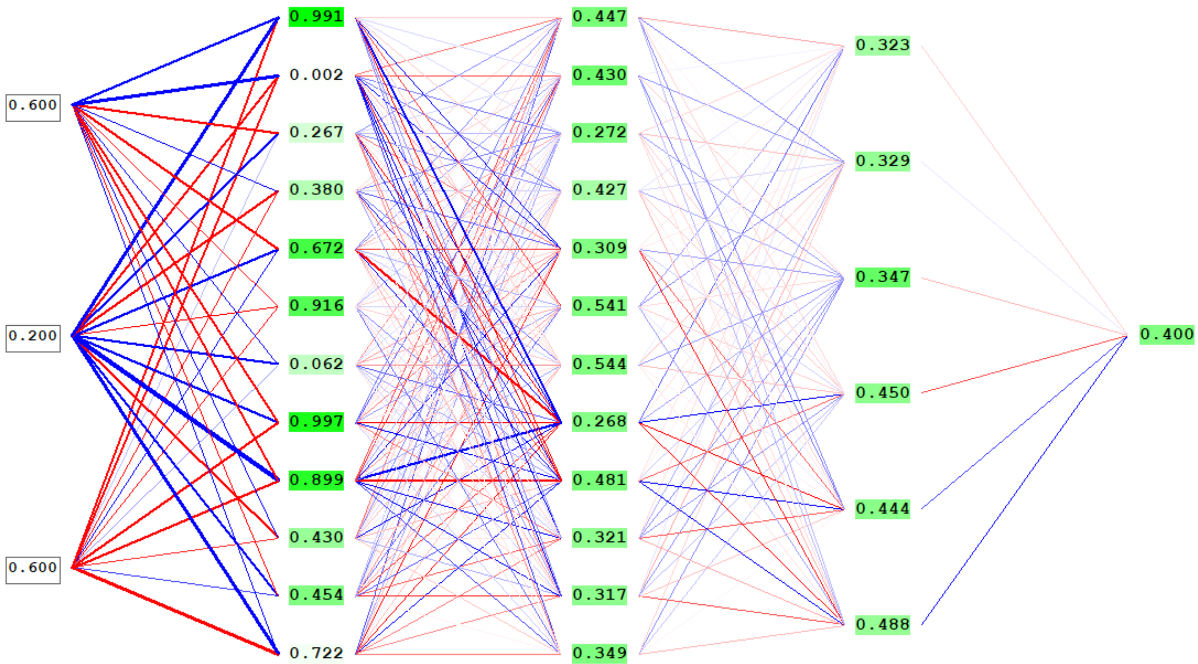

Fig. 3. Neural network model in the NeuroNet environment

To create test models, the training data was created in such a way that it was very similar to real ones (the real temperature and precipitation indicators were used). The goal of this stage is to determine the very possibility of such calculations and choose the architecture of the neural network. To submit the data to the network, they are normalized in the range from 0 to 10.

At the stage of final training, the static data on non-combat losses of the XVIII-XIX centuries is used to clarify the weighting coefficients.

Next, we calculate the number of losses during combat operations. In the Middle Ages, there were two types of such actions: military clashes and siege fortresses. To calculate the former, it is possible to apply the linear Osipov – Lanchester's law [6]:

$$A_0 - A_t = E(B_0 - B_t) \tag{7}$$

where:

A_0 – primary number of units of side A;

A_t – the number of troops remaining in army A at the time of T;

B_0 – primary number of units of Side B;

B_t – number of troops remaining in army B at the time of T;

E – weapon quality ('*E*'xchange Rate) = (striking ability of side *B*'s weapon) + (striking ability of side *A*'s weapon), (striking strength) = (weapon Quality Factor) * (number of units) [4].

However, such a model cannot work in the case of sedimentary military operations or when different branches of the armed forces collide, since it provides for the principle "one soldier kills one other soldier". To calculate the losses in the "heavy horse cavalry – light infantry" collision, the losses among the infantry are significantly higher than among the cavalry, and it is almost impossible to calculate the losses during the storming of fortresses, since there are several options for storming the fortress to die. Namely, the death of soldiers in clashes at the walls of the fortress, the death from starvation in a fortress that is under siege, the death during the storming of the fortress, and so on.

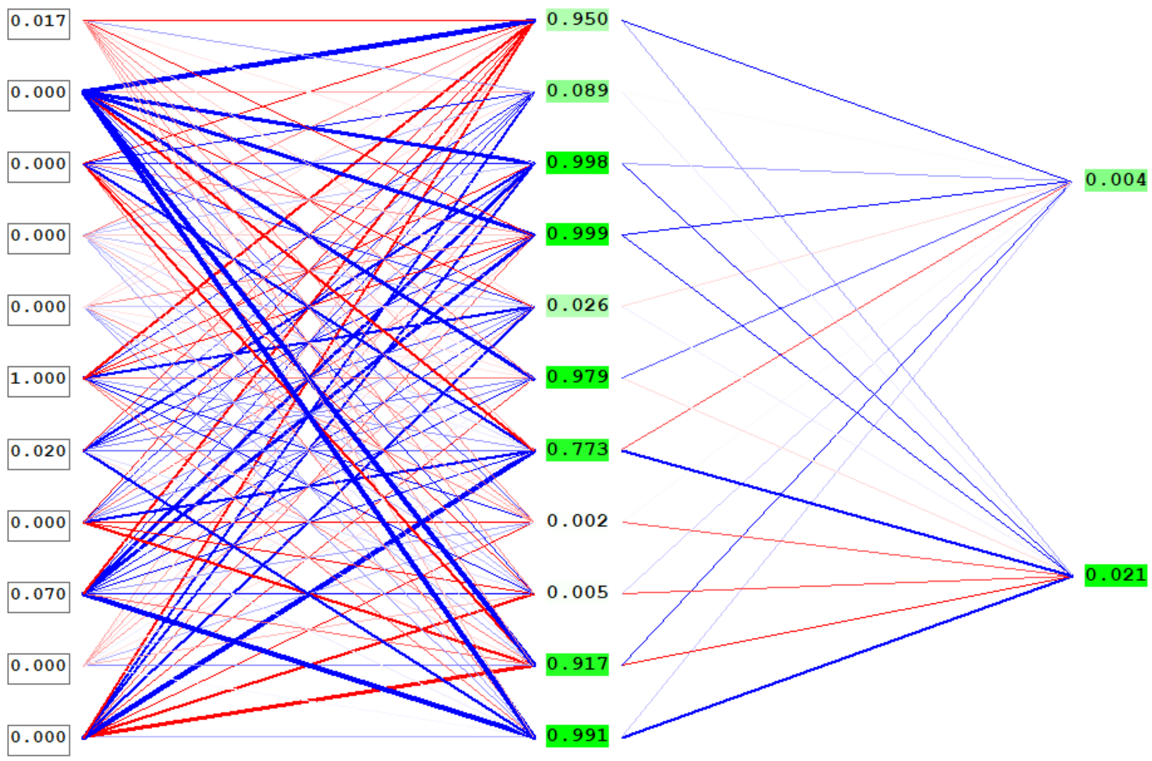

Fig. 4. Neural network model, in the NeuroNet environment

Because of this, it is proposed a less accurate, but a more flexible model for approximate counting of victims of those collisions, the number of victims of which, even approximately, was impossible to estimate using classical methods. Knowing which forces participated in a combat collision, you can calculate approximate combat losses based on the data on losses in similar collisions. Thus, after analyzing a large number of battles, you can deduce some patterns. For such purposes, a direct distribution neural network can be better suited. To clarify the number of inputs, we list all branches of the armed forces: heavy cavalry, light cavalry, infantry, archers, non-professional soldiers (people's rebels, militias, etc.). The result of the battle will be submitted for one more input (1 – the first army won, -1 – the second army won, 0 – the battle ended without any significant results on both sides). Thus, the network has 11 inputs and 2 outputs, and another hidden layer of 11 neurons is also needed (Fig. 4). To train the network, a small training sample was created from the real data taken from open sources. At the outputs, we get the number of losses for each of the conflict sides.

Let's estimate the size of the medieval army. The main criteria by which you can assess the correctness of the assumption for the probable number of troops are:

- column length on the march (S);
- length of the wagon train (S_{wagon});
- main column length (S_{column}).

$$S = S_{wagon} + S_{column} \quad (8)$$

As noted earlier, for normal movement on the march, a horse and a rider needed 6 meters, then (as a rule, the number of rows is taken no more than 2, because the roads in the Middle Ages were quite narrow):

$$S_{column} = \frac{kol \cdot 6}{r} \quad (9)$$

where:

r – number of rows during the transition.

To feed the army per day, you will need m_1 ton of food (based on the calculation that one adult man needs 1 kg of food per day):

$$m_1 = \frac{kol + \sum kol \ i_j \ v_j \ p_j}{1000} \quad (10)$$

where:
i – the number of horses of breed j;
v – the weight of hay that one horse of breed j needs;
p – the percentage of breed j horses in the army.

Thus, knowing the required mass of forage, you can calculate the length of the train using the following formula:

$$S_{Swagon} = \frac{m_1 - a \cdot kol + m_2}{0{,}07} \cdot 8 \quad (11)$$

where:
m_2 – other non-food cargo: siege vehicles, trebuchets, wounded, prisoners, engineering troops.
a – the average weight of the load that is on the horse with the rider.

In this way, you can calculate the length of the column on the march, as well as the time it takes for the rider to travel from the head of the column to the end or vice versa. If the rider cannot cover this distance in a day, the column can be considered unmanageable, that is, it could not exist in reality.

Creating a Web-system

Based on the proposed mathematical models, a WEB system was developed with applications for calculating combat and non-combat losses. The system provides several use cases, namely (Fig. 5):
– calculation of non-combat losses;
– calculation of combat losses;
– total loss calculation;
– checking the number of troops.

Calculating non-combat losses and checking the number involves launching the first browser application, while the task of calculating combat losses is to launch the second browser application. The task of calculating total military losses involves running both applications.

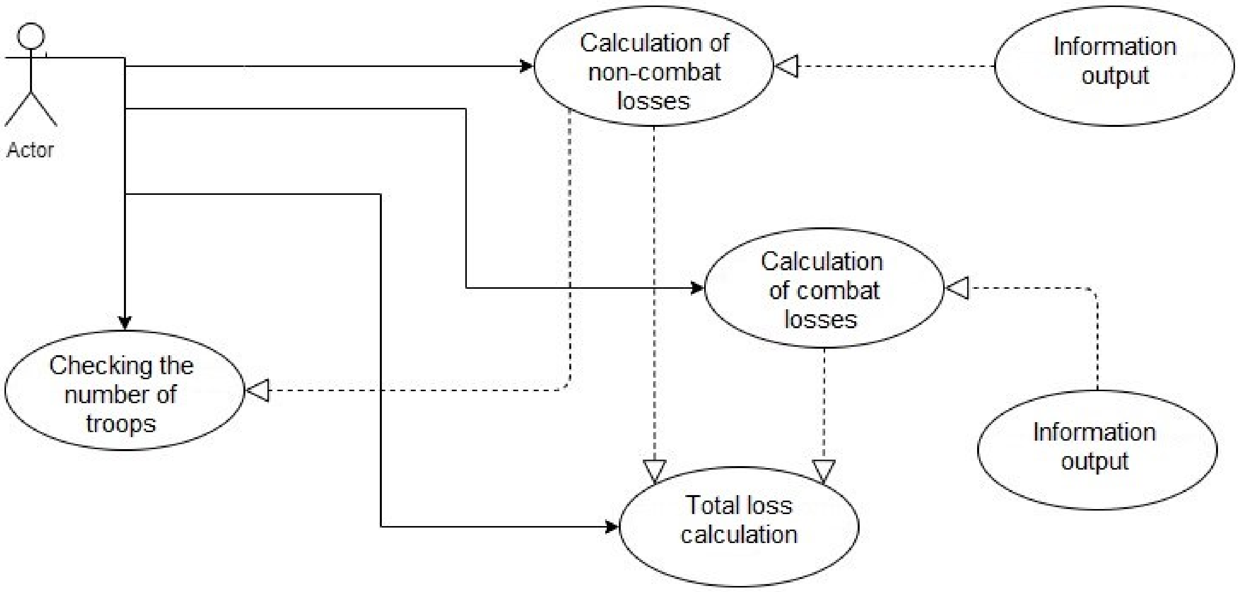

Fig. 5. Diagram of Information System use cases

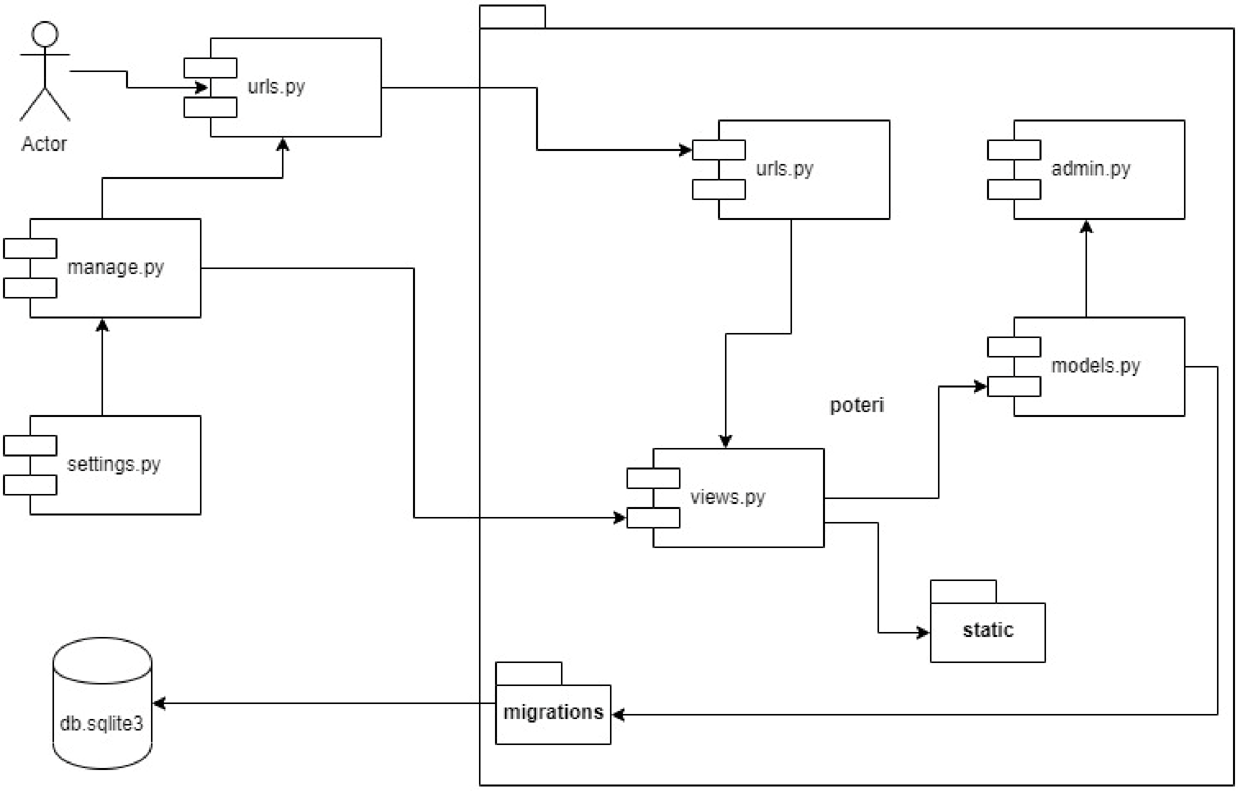

Fig. 6. Component diagram – Django file and folder structure

The Django framework was chosen as the basis for developing the application, Python was chosen for writing browser logic, and the Brython library was chosen as the translator. In the file urls.py the binding URL is located, i.e. they connect the link URL and the function that is called out when requesting this link. The functions themselves are located in the file views.py, they already access HTML templates which are stored in the folder settings.py (Fig. 6).

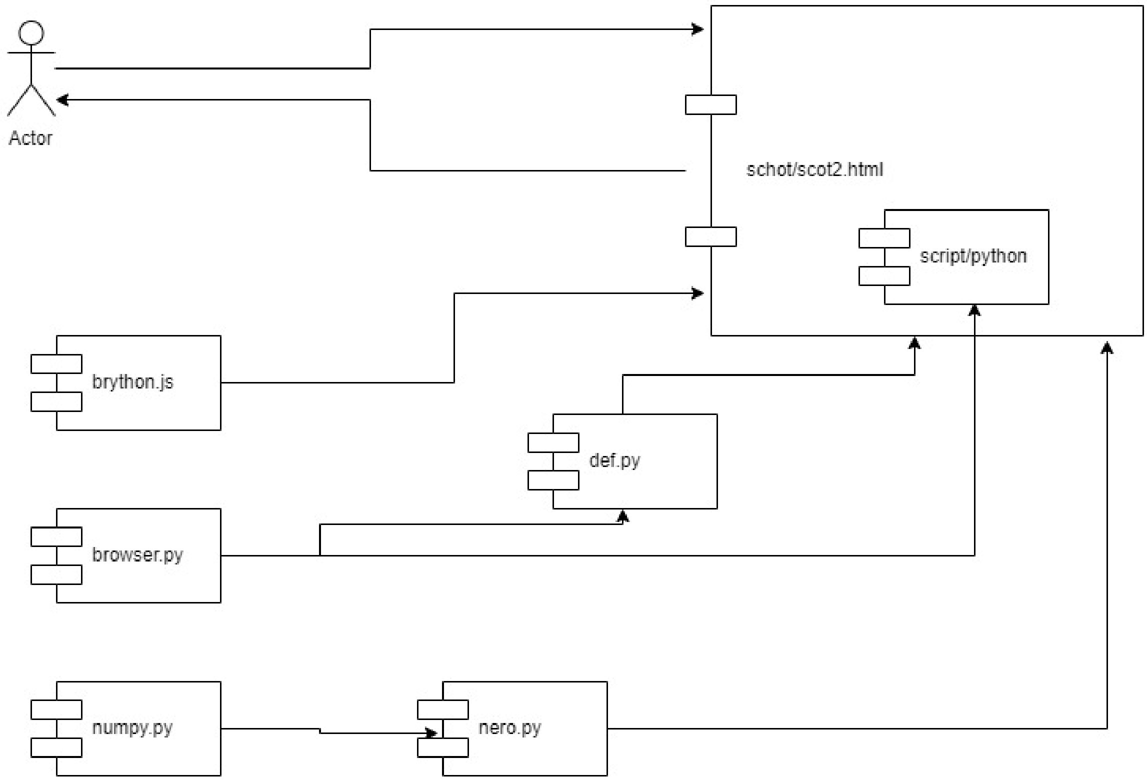

Fig. 7. Component diagram – structure of a browser application for calculating non-combat losses

The code embedded in the HTML file accesses the browser library and also calls out functions from files def.py, where the basic logic of the application is stored, and file nero.py, where the neural network will be located. The Brython file.js is a translator of Python code to JS, browser.py files and numpy.py are Python libraries for managing page elements and for mathematical calculations that will be needed when creating a neural network (Fig. 8, Fig. 9).

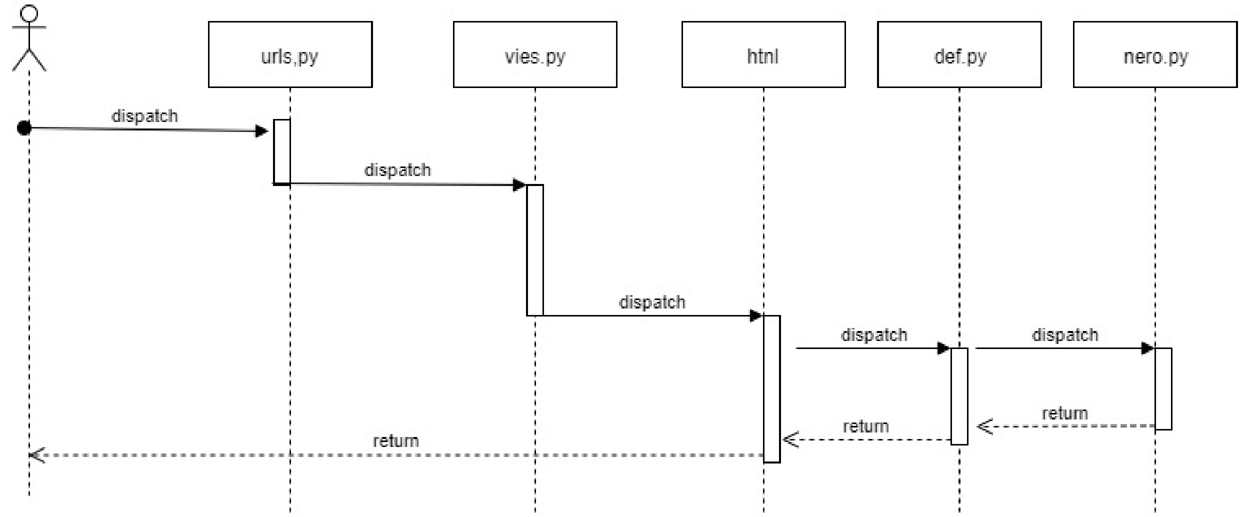

Fig. 8. Sequence diagram, non-combat loss counting process

A special feature of the Brython translator is that all files with code are connected not to each other, but to the main HTML file. Libraries that are embedded in the translator are connected in the usual way, but the modules work as shown in the class diagram (Fig. 9).

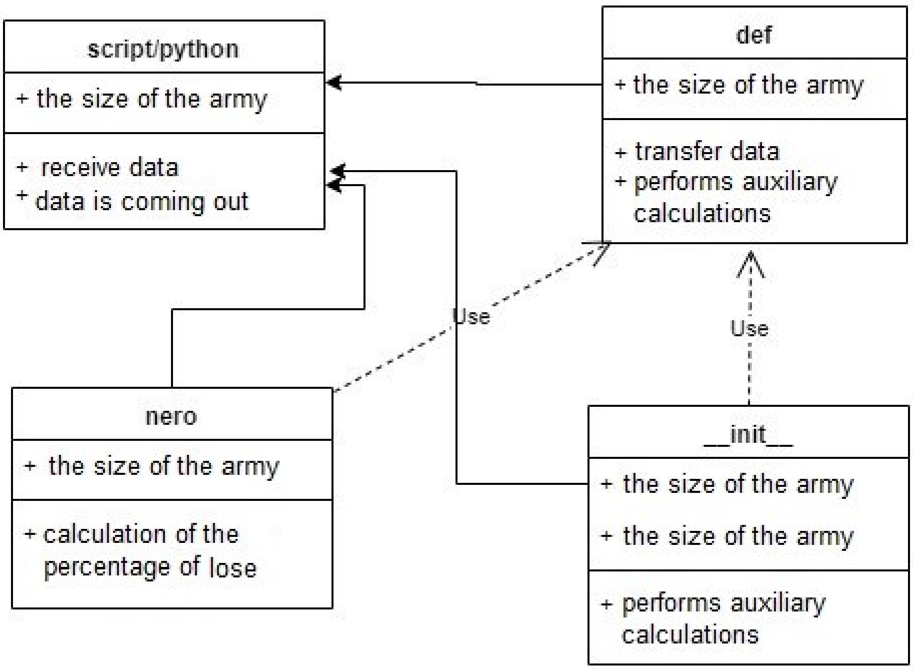

Fig. 9. Class diagram – structure of a browser application for calculating non-combat losses

The second browser application must take data from the user's fields and pass it to calculate the neural network, but in this case a static HTML page will be used (Fig. 10), the files have the same purpose, but this application does not have a script embedded in the HTML file itself.

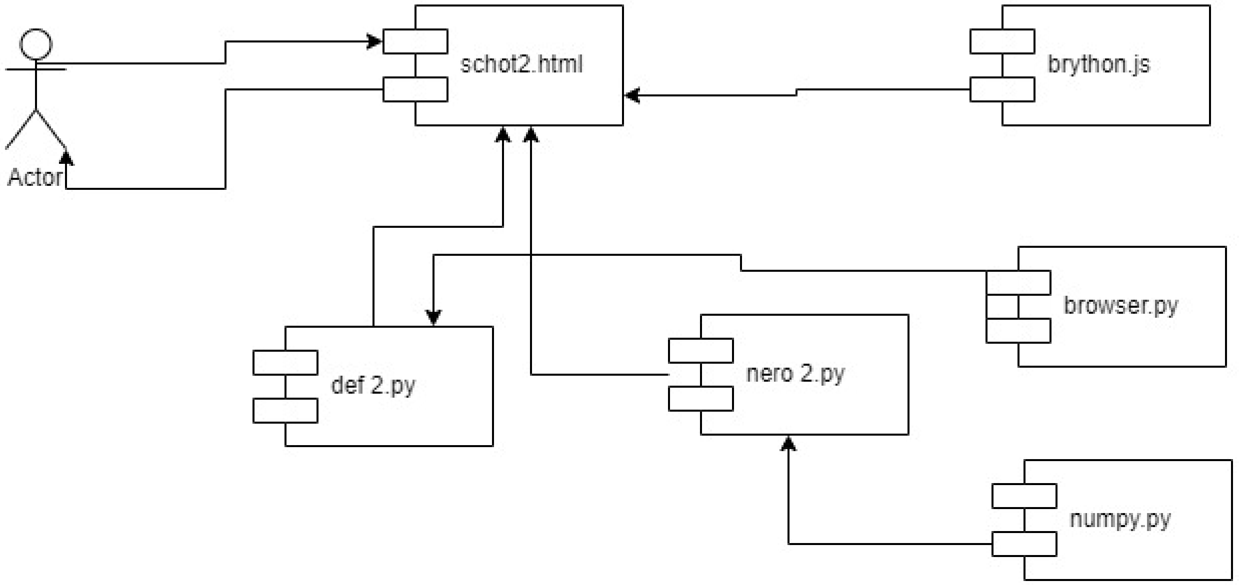

Fig. 10. Component diagram – structure of a browser application for calculating combat losses

In the final version, the system looks like a site that has the following structure: home page (Fig. 11), a project page, non-combat loss calculation tool, combat loss calculation tool, project news.

Fig. 11. Middle part of the site's home page

Since during large military campaigns the armies were divided into several columns, it would be interesting to observe the changes of all the columns at once. Before loading the home page, the user is asked to select the number of columns and the type of Army (Fig. 12).

Fig. 12. Pre-Launch page of the first browser app

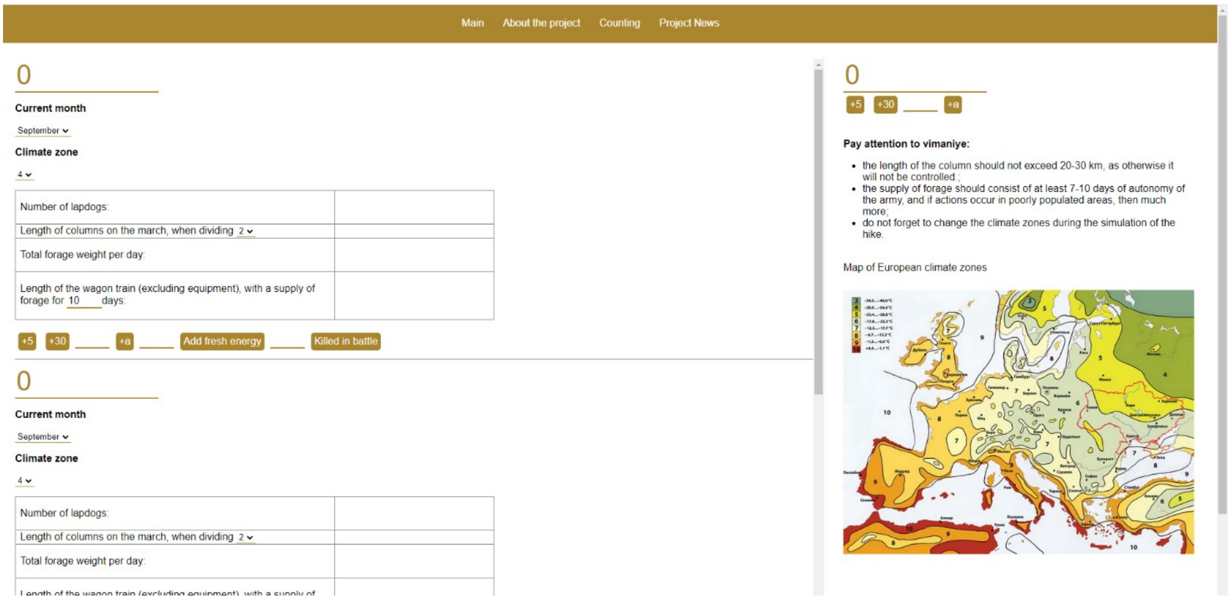

Fig. 13. Calculation of non-combat losses

To implement multiple applications simultaneously (Fig. 13) cycles are used: when rendering a page, the framework copies some sections of code a given number of times and assigns each element in each of the copies its own index, for example:

def func5{{ forloop.counter }}(event):
after the rendering, it will look like this:
def func51(event):

This makes it very easy to manage and create executable code from the server already at the page rendering stage. This made it possible to use each calculation block independently of the others, while leaving the possibility of unified management of all blocks. To facilitate calculations during the operation of the neural network, all weighting coefficients are already feeded in the code.

The user can add any number of campaign days, new forces, or combat losses. The current month switches automatically. the user must monitor the climate zones himself. The neural network has been trained to work only with the climatic zones in Europe.

To calculate combat losses, a neural network has also been developed that can calculate losses in a medieval combat, but it should be noted that we do not have the objective information about medieval losses, so we used loss data calculated by historians using various methods to train it. It should also be pointed out that the neural network does not take into account indirect combat losses (capture, murder of prisoners, murder of civilians, etc.) (Fig. 14)

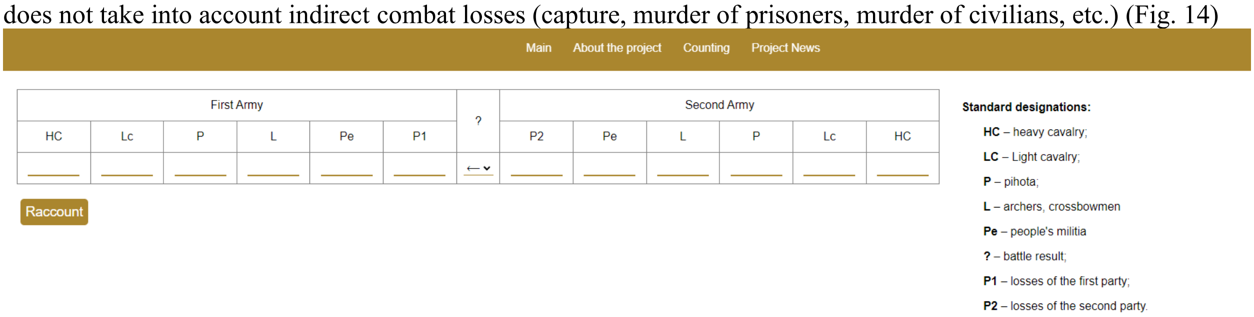

Fig. 14. Calculation of combat losses

Let's calculate the losses of the Mongol army during the eastern campaign: from historical sources, it is known that the Mongols approached our lands in the autumn of 1237, dividing into three columns. Let’s find out what the number of these columns was in a month of the campaign. The total number of troops was approximately 40,000, the campaign was begun in October, the events happened near the cities of Ryazan, Suzdal and Vladimir, which corresponds to the climatic zones 4 and 5 [9, 10]. For 30 days of the campaign, non-combat losses amount to just under 3,000. The length of the largest column (15,000 soldiers from the beginning of the campaign) is 15 km, and 294 tons of food will be needed per day to feed the army. The user makes a decision about the adequacy of the column characteristics independently. The application only performs calculations based on the data fed by the user. According to the modern ideas of historians, the length of the columns could hardly exceed 20 – 30 km.

Conclusions

The project has developed the models of an information system for analyzing the number of troops and calculating military losses of the remote past. Mathematical models have been described, historical sources have been analyzed, and detailed models of the application have been created using the UML modeling language, which allows you to understand its interface in detail at the modeling stage.

Based on the data obtained in the analysis of historical literature, the neural network architectures have been developed to determine non-combat losses in the medieval army and determine combat losses based on the data on the number of each branch's soldiers of the armed forces who were involved during the battle.

Web applications for calculating combat and non-combat losses have been developed, interface design and mini-block for publishing system News have been developed either. The non-combat losses suffered by the Mongol army in the first month of the Western campaign (1236) have been calculated as the example.

References

1. Zhukov K.A. Drevniaia Rus. Ot Riuryka do Batыia / K.A. Zhukov, D.Iu. Puchkov. — S-Pb. : Pyter, 2019. – 288 s.
2. Kuvaldyn S.A. Ystoryia s klymatom. Khymyia y zhyzn 2017 - №4
3. Babych M.V. Ystoryy neboevыkh poter chastei, dyslotsyrovannыkh v Ynhermanlandyy v 1711 hodu / M.V. Babych, Y.V. Babych – Voina y oruzhye. SPb., 2012. Ch. 1. S. 47-55
4. Venttsel E.S. Osnovы teoryy boevoi эffektyvnosty y yssledovanyia operatsyi / E.S. Venttsel, Y. V. Khudiakov, Ya. M. Lykhterev, Yu. H. Mylhram – M.: VVYA, 1961 – 524 s.
5. Uvarov D. Voennыe potery v Srednevekove [Elektronnyi resurs] – URL: http://xlegio.ru/ancient-armies/military-organization-tactics-equipment/medieval-military-losses/ (15.05.2022)
6. Mytiukov N. V. Opredelenye zhertv voin cherez lanchesterskye modely/ N. V. Mytiukov //Ystorycheskaia psykholohyia y sotsyolohyia ystoryy 2009 - №2 – S. 122 – 140
7. Alekseev V. «Y bыst secha tu velyka...»/ V. Alekseev //Tekhnyka molodezhy Moskva, 1988. – №2 – S. 56–57
8. Vazhneishye bytvы Srednykh vekov. Dyletant [Elektronnyi resurs] – URL: https://diletant.media/top-5/3157/ (19.05.2022)
9. Khrapachevskyi R. Velykyi zapadnыi pokhod chynhyzydov na Bulhar, Rus y Tsentralnuiu Evropu [Elektronnyi resurs] – URL: http://xlegio.ru/ancient-armies/medieval-warfare/genghisids-great-western-campaigen/ (29.05.2022)
10. Zhaksыlыk S. O chyslennosty monholskoi armyy v Zapadnom pokhode/ S. Zhaksыlыk //Voprosы Ystoryy y arkheolohyy Zapadnoho Kazakhstana. 2010 – № 1. – S. 55 – 73
11. Kapeleshchuk A.O. Postanovka zadachi vyznachennia chyselnosti viiska u serednovichchia iz vykorystanniam matematychnoho modeliuvannia ta informatsiinykh tekhnolohii // Molodizhna nauka u konteksti suspilno-ekonomichnoho rozvytku krainy: zbirnyk tez dopovidei uchasnykiv V Yuvileinoi uchnivsko-studentskoi konferentsii, Cherkasy, 2 hrudnia 2021 r. – Cherkasy: Skhidnoievropeiskyi universytet imeni Raufa Abliazova, 2021. – S. 214-216
12. Kapeleshchuk A.O, Melnykov O.Iu. Prohnozuvannia chyselnosti viiska u serednovichchia iz vykorystanniam matematychnoho modeliuvannia // Materialy IV Vseukrainskoi naukovo-praktychnoi internet-konferentsii studentiv, aspirantiv ta molodykh vchenykh za tematykoiu «Suchasni kompiuterni systemy ta merezhi v upravlinni»: zbirka naukovykh prats / Pid redaktsiieiu H.O. Raiko. – Kherson: Vydavnytstvo FOP Vyshemyrskyi V. S., 2021. – С.228-230. – ISBN 978-617-7941-55-1 (elektronne vydannia)

Oleksandr Melnykov **Олександр Мельников**	PhD in Technics (Candidate of Technical Sciences), Associate Professor of the Department of Intelligent Decision Making Systems, Donbas State Engineering Academy, Kramatorsk, Ukraine, e-mail: alexandr@melnikov.in.ua https://orcid.org/0000-0003-2701-8051	кандидат технічних наук, доцент, доцент кафедри інтелектуальних систем прийняття рішень, Донбаська державна машинобудівна академія, Краматорськ, Україна.
Andrei Kapeleshchuk **Андрій Капелещук**	student of FAMIT, Donbas State Engineering Academy, Kramatorsk, Ukraine, e-mail: kapeleshchuk.k@gmail.com	студент спеціальності «Системний аналіз» факультету автоматизації машинобудування та інформаційних технологій, Донбаська державна машинобудівна академія, Краматорськ, Україна.