Computer Science • 14 (3) 2013 http://dx.doi.org/10.7494/csci.2013.14.3.423

Grzegorz Jaśkiewicz

GEOLOCALIZATION OF 19TH-CENTURY VILLAGES AND CITIES MENTIONED IN GEOGRAPHICAL DICTIONARY OF THE KINGDOM OF POLAND

Abstract *This article presents a method of the rough estimation of geographical coordinates of villages and cities, which is described in the 19th-Century geographical encyclopedia entitled: "The Geographical Dictionary of the Polish Kingdom and Other Slavic Countries"* [18]*. Described are the algorithm function for estimating location, the tools used to acquire and process necessary information, and the context of this research.*

Keywords Geographical Dictionary of Polish Kingdom and Other Slavic Countries, natural language processing, geolocalization, statistics, information extraction

1. Introduction

"The Geographical Dictionary of the Polish Kingdom and Other Slavic Countries" is an encyclopedic dictionary published between 1880 and 1902 in Warsaw. The book consists of 15 volumes and is a rich source of information about the geography of the Central European region. The main focus of the dictionary is the Polish-Lithuanian Commonwealth [11] and the neighboring countries. In the dictionary, there is information about *i.a.*:

- administrative division – voivodeships, governorates, districts;
- demographic data – population size and structure;
- economic data – agricultural and industrial production rates, fields, factories, financial assets;
- human settlements – cities, villages, colonies;
- bodies of water – rivers, streams, lakes;
- transportation and communication infrastructure – transport routes, railways, trade routes, post offices;
- potentates – village owners, dukes, nobility;
- church administrative divisions – parishes, deaneries;

Much of the information mentioned above also contains a historical aspect, *e.g.*

- how the demographic structure in a particular city changed over the years,
- what major historical events took place in the proximity of the described entities.

The book is written in a Polish dialect spoken in the 19th Century [1], slightly different than the language spoken in modern-day Poland.

Although, the dictionary is no longer being issued, individual volumes are still in circulation and a hardcopy of the dictionary is still available.

2. Related works

2.1. Works in the field of science

This study could be classified into the Geographic Information Retrieval (GIR) field. This is a small and relatively new branch of Information Retrieval which is closely related to Geographic Information Systems. One of the first TREC-style forums, called GeoCLEF[1] [9], was started in 2005 to evaluate GIR systems. The focus area of the TREC workshop are geography-related queries in a document search, which has an important application in the search-engines realm [8], *e.g.* search for documents relevant for query ‘‘`pizza in Warsaw`’’.

Another important application of GIR is georeferencing: a process of assigning geographical coordinates to unstructured textual data. There have been applications of georeferencing to digitialize historical data e.g. [19], [10]. There is known application of probability calculus in order to incorporate uncertainty of location estimation

[1] Cross-Language Evaluation Forum

into georeferencing [8]. In general, georeferencing is assumed to work with arbitrary text in a natural language. The dictionary text has repetitive language patterns, which form some kind of a structure over textual data. In case of the dictionary statistical models could be applied in very direct way.

There is a small amount of available literature on geoparsing Polish documents, and quite possibly, there has been no approach to geoparse the dictionary; so, this work also represents a new contribution to this field.

2.2. The dictionary digitization attempts

There have been several major efforts to digitize the dictionary. Usually, the first step of digitalization of any paper-based manuscripts is scanning. One of the first of such efforts, resulting in a CD-ROM publication of the dictionary, was made by the Polish Genealogy Society of America[2] (abbrev. PGSA).

One of the first digitized versions of the dictionary (available online) was created in 2005 by Dr. Janusz Bień. It is based on DjVu format and is freely available on the internet[3]. His version was supplied with text indices to enhance the search algorithm [3]. Independently, in 2005, PGSA made an effort to run OCR[4] on the previously-scanned text.

In the years 2005–07, research was carried out by Forschungsgruppe Grafschaft Glatz[5]. The researchers focused on translating the dictionary into the German language. This project did not succeed in its original intent, which was a translated dictionary, but it provided an alternative source of the dictionary text in the digital format[6].

The Małopolska Digital Library[7] has a free online copy, which is also DjVu-based. It also provides the text obtained by OCR processing, which was used to create the text-search indices on the library webpage. This source was started in 2006.

Another online version of the dictionary is in the archives of Domain of Internet Knowledge Repository of ICM[8]. Except for the text-search, this webpage also contains an entry-search and a page index. The user can choose to navigate to a selected page of the dictionary or search for a particular dictionary entry.

Around the year 2008, the dictionary was referenced in the Polish Wikipedia. A page with hyperlinks to several of the entries in the dictionary[9] was started on Wikipedia. Many pages in Wikipedia were supplemented by the contents of the dictionary and some are an exact copy of the corresponding dictionary entries.

[2] see: www.pgsa.org

[3] see: http://www.mimuw.edu.pl/polszczyzna/SGKPi/

[4] Optical Character Recognition (see generally [16])

[5] eng. Research group Glatz

[6] refer to: wiki-en.genealogy.net/SlownikGeo for results of PGSA and FGG cooperation

[7] see: http://mbc.malopolska.pl

[8] see: http://dir.icm.edu.pl/pl/Slownik_geograficzny/

[9] see: http://pl.wikipedia.org/wiki/Kategoria:
Skarbnica_Wikipedii/S\%C5\%82ownik_geograficzny_Kr\%C3\%B3lestwa_Polskiego

The dictionary is also an interest of the SYNAT project [2]. The research presented in this article is part of this project. The SYNAT project aims to create an open-hosting repository for assets of Polish science. The dictionary is a good example of such an asset. The project explores many methods to host, extract, and present the knowledge contained within the dictionary. This article presents methods used to extract and process information about the location of human settlements described within the dictionary.

3. Materials and methods

The complete toolbox for the location estimation consists of:

- software for an auxiliary data acquisition,
- parser for processing the dictionary text,
- location estimation algorithm,
- validation engine to test estimation quality,
- data exporters, acting as presentation layer

In this paper, the location estimation algorithm and parser will be discussed in detail, while other parts of the system will be described only briefly.

The algorithm for extracting the locations of the settlements in the dictionary is strongly based on statistical concepts. For this reason, before discussing the software components which constitute the whole system, the main concept of the algorithm for estimating geographical location of the villages will be shown in a formal manner.

3.1. Mathematical concept of location estimation

3.1.1. General concept

The general idea of the algorithm is to analyze the text, try to extract phrases giving clues about a possible location of the settlement, and then to derive probability distribution of the location from each meaningful phrase. This section introduces the nomenclature used in the rest of this article and explain:

- what is a phrase and when a phrase is "meaningful",
- what is a probability distribution for location of a settlement,
- having set of distributions, how the final answer is computed.

The primary input for the estimation algorithm is the dictionary entry for a settlement. This is a sequence of simple phrases which can be single words, numbers, or punctuation marks. Word `‘‘rzeka’’`, `‘‘3’’` and `‘‘południe’’` are examples of simple phrases. The set of all phrases will be denoted as $\mathbb{W}$. The information about phrase order will be described as relation $\succ$, *e.g.* $w_1 \succ w_2$ could be read as phrase w_1 preceeds phrase w_2.

In general, the concept of the algorithm could possibly be applied to any entities for which a geographical location could be assigned. These are usually called landmarks. However, full functionality of the algorithm is heavily based on the assumptions which are valid for settlements only (*i.e.* hierarchy induced by the administrative division). So for sake of simplicity, the algorithm will be described to operate on villages, even if the definitions introduced in this section could be extended to any landmarks with a textual description. The set of villages will be denoted as $\mathbb{V}$.

The last component to be introduced is a geographical location. In many practical applications, it is implemented as a latitude/longitude coordinate pair. But, for mathematical elegance, it will be described as a point on the unit sphere $\mathbb{S}_2$[10].

The main effort of this work is to produce an assignment from village space to location space:

$$T : \mathbb{V} \to \mathbb{S}_2 \tag{1}$$

Only entries describing settlements are of interest in this research, so the dictionary could be understood as a function $\mathfrak{D}$ from village space to power space of simple phrases (omitting entries for other entity types, *e.g.* rivers).

$$\mathfrak{D} : \mathbb{V} \to \mathbb{P}(\mathbb{W}) \tag{2}$$

For each occurrence of any word in the dictionary, there exists exactly one simple phrase, thus any word *e.g.* ``wieś'' could map to many phrases in $\mathbb{W}$ as it occurs quite frequently. However, a simple phrase belongs to exactly one dictionary entry, so it holds

$$V_1 \neq V_2 \Rightarrow \mathfrak{D}(V_1) \cap \mathfrak{D}(V_2) = \emptyset \tag{3}$$

Therefore, the "inverse" mapping $\mathfrak{D}^{\text{-}1}$ is well-defined:

$$\begin{gathered} \mathfrak{D}^{\text{-}1} : \mathbb{W} \to \mathbb{V} \\ \forall_{w \in \mathbb{W}}\ w \in \mathfrak{D}\left(\mathfrak{D}^{\text{-}1}(w)\right) \end{gathered}$$

The text processing builds complex phrases out of the other phrases. Thus, the result of text processing could be described as relation Γ:

$$\Gamma \subseteq \mathbb{W} \times \mathbb{W} \tag{4}$$

If $\Gamma(w_1, w_2)$ holds, it indicates that phrase w_2 is part of phrase w_1. By its definition, simple phrase w_{simple} satisfies the following property:

$$\neg\exists_{w \in \mathbb{W}}\ \Gamma(w, w_{simple}) \tag{5}$$

A final phrase w_{final} is a phrase, which is not part of any other phrase, *i.e.*

$$\neg\exists_{w \in \mathbb{W}}\ \Gamma(w_{final}, w) \tag{6}$$

[10] actually the Earth is an ellipsoid, which also could be chosen as the location space model, *e.g.* WGS84 [4]

As mentioned above, every complex phrase is built up from consecutive phrases, so following property holds

$$w_1 \succ w_2 \succ w_3 \ \wedge \ \Gamma(w_1, v) \ \wedge \ \Gamma(w_3, v) \ \Rightarrow \ \Gamma(w_2, v) \tag{7}$$

The order of complex phrases could be inferred from phrases which are part of the complex phrase, *i.e.* for phrase w

$$w_p \succ w \Leftrightarrow \neg\exists_{w_i \in \mathbb{W}} \forall_{w_j \in \mathbb{W}} \Gamma(w_j, w) \ \wedge \ w_p \succ w_i \succ w_j \tag{8}$$

For fixed set of phrases $P \subset \mathbb{W}$ extension is P_e minimal subset of $\mathbb{W}$ which satisfies

$$\begin{aligned} &\forall_{w_2 \in \mathbb{W}} \left[\left[\forall_{w_1 \in \mathbb{W}} \Gamma(w_1, w_2) \wedge w_1 \in P_e \right] \ \Rightarrow \ w_2 \in P_e \right] \\ &P \subseteq P_e \end{aligned}$$

Such a definition may yield some similarities to the construction of the least Herbrand model in the logic programming [13] – in fact, section 3.3 will show a parser, which acts as a rule-based system.

Thus, the text processing on the set P is a process of computing the extension P_e. The process will be indicated by the function

$$T_w : \mathbb{P}(\mathbb{W}) \to \mathbb{P}(\mathbb{W}) \tag{9}$$

The next step of the algorithm is to assign location distributions for all final phrases. This process will be denoted as T_σ

$$\sigma : \mathbb{S}_2 \to \mathbb{R} \qquad \int_{\mathbb{S}_2} \sigma \cdot \partial S = 1 \tag{10}$$

$$T_\sigma : \mathbb{W} \to \sigma \tag{11}$$

The rationale of assigning the distributions of possible location takes into the account spatial relations, which are described by phrase while estimating the position, *e.g.* (see Fig. 1). If a phrase gives no information about village location, it is assigned by T_σ with uniform distribution over $\mathbb{S}_2$ – which is interpreted as "anywhere on the Earth", however such phrases should preferably be omitted in most cases.

Hence, the position P_v of the village V could be estimated as the expected value of normalized product of spatial distributions[11]. A set of distributions for an entry v is computed as follows:

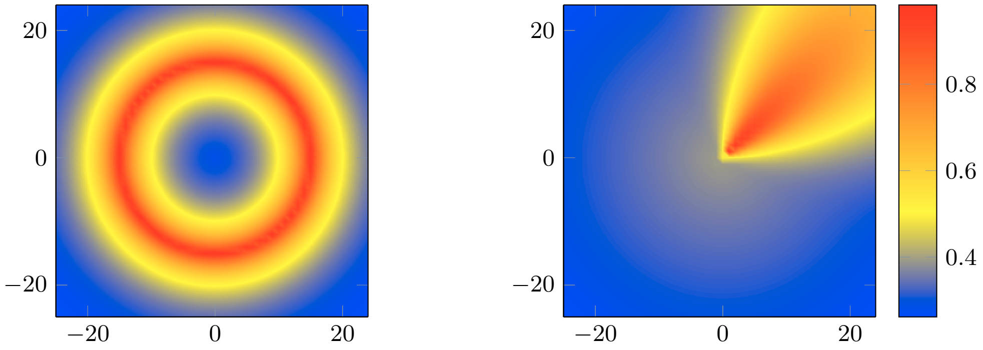
$$V_\sigma = \{\sigma : v \in T_w(\mathfrak{D}(V)) \ \wedge \ T_\sigma(v) = \sigma\} \tag{12}$$

Based on above equation the location for v is

$$E_v = \frac{\int_{\mathbb{S}_2} S \cdot \prod_{\sigma \in V_\sigma} \sigma \cdot \partial S}{\int_{\mathbb{S}_2} \prod_{\sigma \in V_\sigma} \sigma \cdot \partial S} \tag{13}$$

A sample outcome of the algorithm is presented in the Figure 2. The red dots symbolize villages, blue dot – estimation result and green – real location of the village.

[11] technically it's a projection of expected value on $\mathbb{S}_2$



Distribution for phrase: "`10 km from` V_1" Distribution for phrase: "`north-east from` V_1"

Figure 1. The graphical representation of possible location distribution example based on phrases.

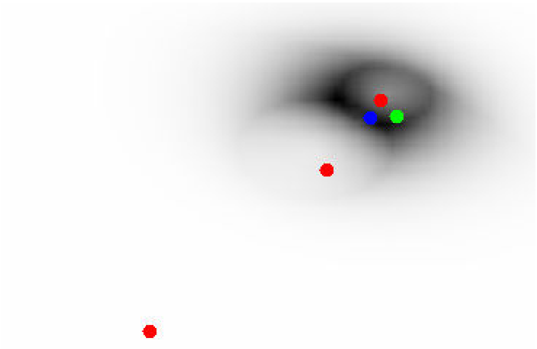


Figure 2. A sample outcome of the algorithm: location distribution the village based on set of settlements and the estimation result.

3.1.2. Setting specific to the research

Typically, geoparsing algorithms operate on additional data describing known facts about geographical objects and spatial relationships between them. These data sources are called gazetteers [5]. The location estimation, which was the aim of the research presented, is based on two types of external knowledge:

- district index: locations of capital cities in districts,
- city index: locations of other settlements – more is better.

Those two indices are modeled as Ξ and Θ – the relations between phrases and locations:

$$\Xi \subseteq \mathbb{W} \times \mathbb{S}_2 \qquad \Theta \subseteq \mathbb{W} \times \mathbb{S}_2 \tag{14}$$

If $\Xi(w,p)$ holds, it indicates that a phrase w could be matched through the city index to a settlement with a position p. Whereas, if $\Theta(w,p)$ holds, it indicates that a phrase w is recognized by the district index as a district with a capital city with a position p.

The algorithm analyzes phrases which indicate city names, relative positions of city names, and district names. Each phrase could be matched to multiple locations through the city index. Those locations are assigned weights based on the proximity of capital cities of districts mentioned in the description.

Both variants of the research were based on a family of gaussian functions ($\mathfrak{G}$) in space with metric $||\cdot||$:

$$\mathfrak{G} \quad \equiv \quad g_{w,s^2}(x) = \frac{1}{\sqrt{2\pi \cdot s^2}} \cdot \exp\left(-\frac{||x-w||^2}{2\cdot s^2}\right) \tag{15}$$

In one variant, distributions were gaussian functions; in the other variant (which used information about spatial relations), distributions were modified gaussian functions (see section 3.4).

The product of the two gaussian functions is a gaussian function with following parameters.

$$g_{w,s^2} = g_{w_1,s_1^2} \cdot g_{w_2,s_2^2} \tag{16}$$

$$w = \frac{s_1^2 w_1 + s_2^2 w_2}{s_1^2 + s_2^2} \tag{17}$$

$$s^2 = \frac{s_1^2 s_2^2}{s_1^2 + s_2^2} \tag{18}$$

The entire counter-domain of mapping T_σ in the first approach is a family of gaussian functions (15). Taking into account the properties (16)–(18), the problem stated in terms of gaussian functions is simplified to calculating the weighted average of the locations of the settlements. In this case, variances of individual functions are treated as weights in the average. The properties (16)–(18) did not hold for functions, used in second variant, so in that case, location was computed with help of discretization.

Only final, non-simple phrases were assigned indicative distributions. Each of those phrases could be matched to many possible villages by the city index.

$$V(w_V) = \{v : \Xi(w_V, v)\} \tag{19}$$

Each of those matched villages contribute an individual gaussian function to location distribution for phrase, *i.e.*

$$T_\sigma(w_V) = \prod_{v\in V(w_V)} g(v, w_V)$$

A function

$$g : \mathbb{S}_2 \times \mathbb{W} \to \mathfrak{G} \tag{20}$$

assigns gaussian function (15) for village matched to phrase in context of processed village. The parameters for the resulting function are chosen as follows:

- w – is position obtained by using the city index,

- s^2 – is chosen based on a relevance of the possible guess.

The relevance score is based on a proximity of capital cities of districts indicated by phrases found in the description. The rationale of this decision is as follows: districts in descriptions are recognized well, and for each entry, there is almost always information about a district. In this paper, the variance s^2 is also dependent on individual villages matching that phrase. For parameters $p \in \mathbb{S}_2$ and $w \in \mathbb{W}$, the function g (20) is created in the following way:

$$V_w = \mathfrak{D}^{-1}(w) \tag{21}$$

$$U_c = \{V_c : w \in T_w\left(\mathfrak{D}(V_w)\right) \wedge \Theta(w, V_c) \wedge w \text{ is final}\} \tag{22}$$

$$v_m = \min_{V_c \in U_c} ||\, p - V_c\, || \tag{23}$$

$$s_v^2 = \frac{1}{\operatorname{card}(V(w))} \cdot f_m(v_m) \tag{24}$$

where f_m is a “falloff” function

$$f_m : \mathbb{R} \to \mathbb{R} \tag{25}$$

Investigating the influence of different falloff functions on the estimation quality was one of the goals of this research.

Example

Consider a simplified entry describing a fictional city V_p:

`“Vp, district X, between V3 and V4, near V5”`

Following conditions holds

$$\begin{array}{rr}
\Xi(\texttt{"V3"}, V_3') & \Xi(\texttt{"V3"}, V_3'') \\
\Xi(\texttt{"V4"}, V_4') & \Xi(\texttt{"V4"}, V_4'') \\
\Xi(\texttt{"V5"}, V_5') & \Xi(\texttt{"V5"}, V_5'') \\
\Theta(\texttt{"district X"}, V_1) &
\end{array}$$

The spatial relation between the cities matched by Ξ and Θ relations are shown in Figure 3.

Therefore, having

$$\begin{array}{ll}
w_3' = \dfrac{1}{2} \cdot f_m(||V_1 - V_3'||) & w_3'' = \dfrac{1}{2} \cdot f_m(||V_1 - V_3''||) \\
w_4' = \dfrac{1}{2} \cdot f_m(||V_1 - V_4'||) & w_4'' = \dfrac{1}{2} \cdot f_m(||V_1 - V_4''||) \\
w_5' = \dfrac{1}{2} \cdot f_m(||V_1 - V_5'||) & w_5'' = \dfrac{1}{2} \cdot f_m(||V_1 - V_5''||)
\end{array}$$

the position of the village V_p is estimated as

$$V_p = \frac{w_3' \cdot V_3' + w_3'' \cdot V_3'' + w_4' \cdot V_4' + w_4'' \cdot V_4'' + w_5' \cdot V_5' + w_5'' \cdot V_5''}{w_3' + w_3'' + w_4' + w_4'' + w_5' + w_5''}$$

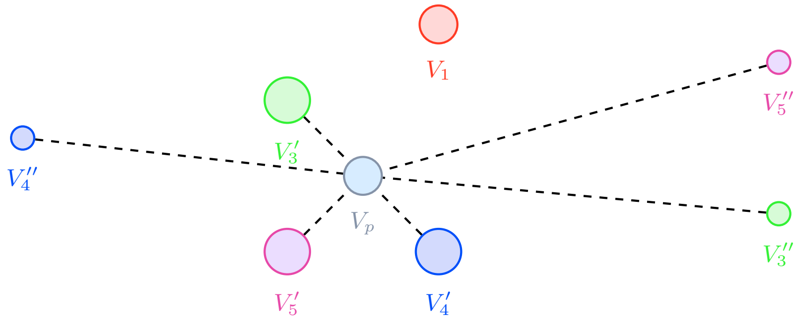


Figure 3. The relative position of cities: V_1, V_3', V_3'', V_4', V_4'', V_5', V_5'' and the estimation of location of the village V_p.

3.2. Data aquisition

In the SYNAT project, two data sources with the dictionary text were acquired. Those sources were PGSA and ICM (see section 2.2). In both cases, there were errors introduced by an OCR algorithm. One of the tasks of the SYNAT project is to implement effective OCR techniques for text digitalization. Much effort was put to improve quality of OCR outcomes for the dictionary [6]. This research, however, was conducted under the assumption of the fact that text data is almost error-free. Such data was obtained by the human labor. Volume II of the dictionary, from PGSA, consisting of approx. 800 pages, was corrected by hand by Mrs. Ewa Wiszowata. The sample provides approx. 8300 entries and was used as an input to the location estimation algorithm.

The dictionary is written in an encyclopedic style: there are many common phrases, and sentence structures for different entries share many common features.

The entries usually provide information in a semi-structured fashion (see Fig. 4):

1. Name of the locality.
2. Type of the locality.
3. District.
4. Parish.
5. Population figures, agricultural data, number of houses, distances from other localities.
6. Other data: ownership, historical events, etc.

In section 3.1.2, two external knowledge sources were described. Both of them were constructed with a help of Wikipedia. The city index was constructed with help from the pages called "Treasury of Wikipedia". The purpose of those pages is to provide facts, which are helpful in writing new articles on Wikipedia. One such page in the Polish Wikipedia contains information about the geographical locations of modern-day medium- to large-sized cities in Poland. This source is very rich in information; however, there are two major drawbacks:

$$\underbrace{\textbf{Derewiancze,}}_{\text{name}}\underbrace{\texttt{wś,}}_{\text{type}}\underbrace{\texttt{pow. ostrogski,}}_{\text{district}}\underbrace{\texttt{od m. pow. Ostroga o 6 w. oddalona}}_{\text{relative distance}}\ \ldots$$

$$\underbrace{\textbf{Derewiany,}}_{\text{name}}\underbrace{\texttt{wś,}}_{\text{type}}\underbrace{\texttt{pow. uszycki,}}_{\text{district}}\underbrace{\texttt{gm, i par. Kitajgród,}}_{\text{district and parish}}\underbrace{\texttt{177 dusz męz.}}_{\text{population size}}\ \ldots$$

$$\underbrace{\textbf{Derewiczna,}}_{\text{name}}\underbrace{\texttt{wś,}}_{\text{type}}\underbrace{\texttt{pow. radzyński,}}_{\text{district}}\underbrace{\texttt{gm. Brzozowy-Kąt,}}_{\text{district}}\underbrace{\texttt{par. Komarówka,}}_{\text{parish}}\ \ldots$$

Figure 4. Sample entries in the dictionary.

- the source contains information about cities which lie within current borders of Poland[12], whereas the Polish-Lithuanian Commonwealth had significantly different borders;
- reliability of this source is a bit questionable as the treasury of Wikipedia seems to be, unlike the regular Wikipedia, prone to acts of vandalism, *e.g.* there could be found entries indicating non-existent cities like "Gura-Kal'var'ya" (in place of "Góra Kalwaria").

These problems limit the possible usability of the algorithm based on it. Thus, the source is an interim solution, acceptable for the sake of constructing a proof-of-concept.

The second external data source is the district index, which is also constructed with help of Wikipedia. There are about 400 different districts in modern-day Poland as well as former districts which existed over a timespan between commonwealth times and modern times. The data was acquired by a `HtmlUnit` webcrawler, which visited Wikipedia pages of Polish districts, searched for capital cities, and extracted their locations. Links to districts were obtained by category and metapages for keyphrases, "former districts", and "Polish administrative division". In individual cases, the webcrawler failed to get information about the capital city. In such cases, information was retrieved manually.

The problems of this data source are as follows:

- district position estimation is based on its capital city position, which sometimes fails to be a good estimation;
- it does not capture administrative division changes over the course of time.

Despite that, the district-location estimation is good enough, even if the borders of certain districts and their corresponding capital cities had changed, it does not introduce a large error into calculations.

[12] there is similar index for Ukraine, but the foreign language was an obstacle for acquiring this source of data

3.3. Parser

The text parsing was described in section 3.1.1 as an abstract relation Γ. In this chapter, it will be explained how complex phrases are produced from a set of simple ones. In other words, a concrete form of Γ relation used in the research will be provided. A classical parsing is based on grammar which follows some strict rules. In natural language processing, parsing deals with disambiguations and multiple interpretations of a single sentence; hence, utilizing rules alone is not sufficient. The data in the dictionary is specific: it is a natural language, but it is written using repetitive phrases and structures. The dictionary parser still must deal with disambiguation, as some basic phrases can have different meanings, *e.g.* ``m.'' is an abbreviation, which can be interpreted, depending on phrase context, as:

1. ``miasto'' – city,
2. ``metr'' – metrical unit,
3. ``morga'' – area unit,
4. ``mężczyźni'' – population, men.

In the presented example, both ``miasto'' and ``metr'' are relevant phrases for location estimation. In fact, all the presented interpretations are interesting for the purposes of the SYNAT project, because the parser is constructed as a solution which would extract different types of information not necessarily related to the location estimation, *e.g.* demographical data.

The parser operates on grammatical rules in form

$$P \leftarrow m_1(P_{k+1}), \quad m_2(P_{k+2}), \quad \dots, \quad m_n(P_{k+n}) \tag{26}$$

where:

$$\begin{aligned} P_{k+1}, P_{k+2}, \dots, P_{k+n} \in \mathbb{W} &\quad \text{– are consecutive (w.r.t } \succ\text{) phrases} \\ m_1, m_2, \dots, m_n \in \mathbb{M} &\quad \text{– are match functions} \end{aligned}$$

The match functions describe conditions in which a phrase must fulfill in order to be matched by parser rules. It is a Boolean function over the phrases space

$$m : \mathbb{W} \to \{T, F\} \tag{27}$$

After the successful application of rule (26), a set of existing rules is extended by a new phrase P. The new phrase succeeds phrase P_k and precedes P_{k+n+1}. This extension conforms to the invariant of the $\succ$ relation (7). The process could be understood as forward-chaining reasoning [15], where grammar rules correspond to inference rules. A visualisation of the parsing output for a sample sentence has been shown in Figure 5.

The parser uses the concept of a phrase type. Phrases can have multiple types. Let $T_{\mathbb{W}}$ be a set of phrase types and t be a relation to determine phrase type

$$t \subseteq \mathbb{W} \times T_{\mathbb{W}} \tag{28}$$

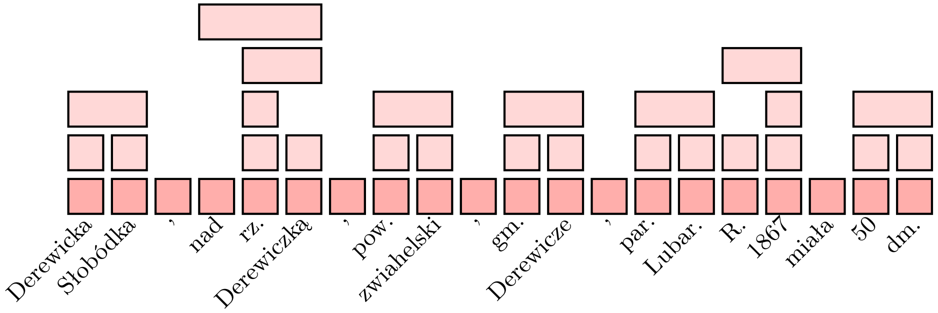


Figure 5. Visualisation of parsing output for the sentence: ``Derewicka Słobódka, nad rz. Derewiczką, pow. zwiahelski, gm. Derewicze, par. Lubar. R. 1867 miała 50 dm.''. Dark red squares represent simple phrases, light red – complex phrases.

Inheritance relation could be introduced on phrase types (is-a relation known in the object-oriented programming [14]). The reason to use it was to construct parser rules, which would operate on hyponyms [7]

$$\rhd \subseteq T_{\mathbb{W}} \times T_{\mathbb{W}} \tag{29}$$

The relation is reflexive and transitive [12]. It will be used in infix notation. $a \rhd b$ is understood as a is kind of b, *i.e.* $a = b$ or a is subtype of b . Let $\overrightarrow{\rhd}$ be the transitive closure of $\rhd$.

In the following research, match functions were used:

1. basic phrase match by regular expression (POSIX style [17]),
2. stemmed phrase text equality,
3. phrase type equality,
4. phrase subtype relation,
5. metafunctions: logical compositions of the above.

The match condition 4 is satisfied for phrase p and type t_p if it holds:

$$\exists_{t \in T_{\mathbb{W}}}\ t(p, t) \wedge t \overrightarrow{\rhd} t_p \tag{30}$$

The examples of all introduced 1–5 match conditions will be presented below.

Example

Let consider a text: ``32 kilometry od m. Zgierza''[13] and a set of the following rules:

$$\texttt{integer} \leftarrow \mathit{regexp}(\texttt{[0-9]}^{+}, P_1) \tag{31}$$

$$\texttt{metrical-unit} \leftarrow \mathit{stem}(\texttt{``kilometr''}, P_1) \tag{32}$$

[13] eng. *32 kilometers form Zgierz city*

$$\texttt{distance} \leftarrow \begin{array}{l} \mathit{subtype}(\texttt{number}, P_1) \\ \mathit{type}(\texttt{metrical-unit}, P_2) \end{array} \tag{33}$$

$$\texttt{city} \leftarrow \mathit{or}(\mathit{regexp}(\texttt{m.}), \mathit{stem}(\text{``}\texttt{miasto}\text{''}), P_1) \tag{34}$$

$$\texttt{name} \leftarrow \mathit{regexp}([\texttt{A-Z}]\,[\texttt{a-z}]^{+}, P_1) \tag{35}$$

$$\texttt{named-city} \leftarrow \begin{array}{l} \mathit{type}(\texttt{city}, P_1) \\ \mathit{type}(\texttt{name}, P_2) \end{array} \tag{36}$$

$$\texttt{relative-position} \leftarrow \begin{array}{l} \mathit{type}(\texttt{distance}, P_1) \\ \mathit{regexp}(\text{``}\texttt{od}\text{''}, P_2) \\ \mathit{subtype}(\texttt{named-object}, P_3) \end{array} \tag{37}$$

Examples of the match condition 1 are presented in the match rules (31), (35), (37) and as a part of the composition rule (34). The stemming match conditions 2 are shown in the rules (32) and as a part of the composition rule (34). The type equality match conditions 3 are presented in the rules (33), twice in (36) and in (37). The subtype relation match conditions 4 are shown in the rules (32) and (37). The rule (34) is a meta match condition 5.

In this example, a following type inheritance dependency is given:

$$\texttt{integer} \rhd \texttt{number} \qquad \texttt{named-city} \rhd \texttt{named-object}$$

The output of the parsing for the presented setting is shown in Figure 6.

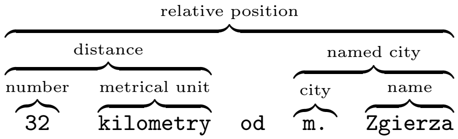


Figure 6. The output of parsing of the example text – complex phrases structure.

3.4. Location Estimation

In the experiment there were considered two forms of the T mapping (11):

1. based on gaussian functions and different types of weight functions f_m (25),
2. based on distributions taking into account information about spatial relation implied by phrases.

In setting 1, the following falloff functions (25) were considered:

- $f_m \equiv 1$ (no falloff),
- $f_m(d) = \frac{1}{1+\ln(1+d)}$,
- $f_m(d) = \frac{1}{1+d^2}$,
- $f_m(d) = \frac{1}{1+d^3}$,
- $f_m(d) = \exp(-d^2)$.

In this setting, location was estimated by calculating weighted average (by (16)–(18)).

In setting 2, the following spatial distribution types were used

- proximity of other village v,

$$\exp(-\sigma \cdot ||x - v||) \tag{38}$$

- proximity of other village v with distance constraint d,

$$\exp(-\sigma \cdot |\, ||x - v|| - d\,|) \tag{39}$$

- relation induced by information about azimuth between villages,

$$\exp(-\sigma \cdot ||x - v||) \cdot \theta(\alpha) \tag{40}$$

  where α is falloff function and α is difference between azimuths in polar coordinates,
- explicit information about geographical coordinates[14].

In this setting, a location was estimated by discretizing a subset of $\mathbb{S}_2$ and calculating the expected value over the interpolation of distribution induced by the discretization.

3.5. Validation

The validation set was prepared manually as a list of 50 dictionary entries describing a village with known location. The entries were selected to have 1–3 phrases that could be recognized as spatial tokens by the parser. Villages described by entry in the validation set were usually medium in terms of population size and had their own page on Wikipedia. The page was read in order to extract village location. Village names in the set were obfuscated in order to not be matched by the city index, which usually contained their precise location.

The error was defined as Earth surface distance in kilometers between the estimated position and the real position. Each experiment run used different settings of the location estimation algorithm. Two metrics were tracked for each of experiment runs:

- average error,
- median of errors.

4. Results and conclusions

All of the software in the presented toolchain was implemented with Java programming language. Software used utility libraries for string processing, collections and I/O operations known as *Apache Commons*[15]. Despite some dialect discrepancies between text in dictionary and modern language, Morfologik[16] stemmer was applied

[14] very rare and mostly available for big cities

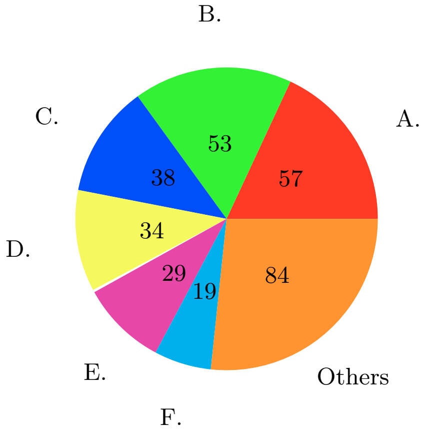
[15] see: `commons.apache.org/`

[16] steamer for Polish language, see: `http://morfologik.blogspot.com/`

successfully. All tests presented in this paper were run on MacBook Pro with 2.4 GHz processor and 8GB of RAM (JVM stack size was capped to 2 GB). Detailed performance tests were not part of this study, however running time was satisfactory ranging from 0.05 s–0.35 s per entry considering middle 95% of execution times. Execution time per entry was proportional to entry size.

The parser was equipped with different rules for gathering different types of data listed in section 1. The number of those rules and their data category is shown in Figure 7. This set of rules resulted in 58.31% total text coverage, *i.e.* 58.31% of simple tokens were part of complex tokens. Average per dictionary entry coverage was 71.39% – many entries were matched in full, while few had very low coverage. There were identified two typical types of entries with low coverage:

1. entries with descriptions of historical events,
2. descriptions of river flows.



A. Economical infrastructure
B. Spatial relations
C. Numbers, time and metric units
D. Settlements names
E. Names and synonyms
F. Administrative division

Figure 7. The number of parser rules per complex token category.

Figure 8 shows how common the information is about districts among dictionary entries. It could be observed that entries with exactly one district token are most common. Entries are missing district information for the following reasons:

- not corrected OCR errors;
- discrepancies between administrative division unit naming in different regions of the Commonwealth, *e.g.* `"Dergacze, gub. charkowska, ..."` where `gub` is governorate, a different type of division unit;

- some entries provide information just about synonymy, *e.g.* `"Derisno, ob. Dzierzazno"`.

It was expected that there will be one district token per entry, reasons for the appearance of multiple district tokens in single entry are following:

- descriptions of entities, which spans through multiple districts, *e.g.* rivers;
- disambiguations – one entry text has multiple distinct subentries, *e.g.* multiple villages with same name;
- descriptions of historical events related to settlements referering to other regions of the Commonwealth.

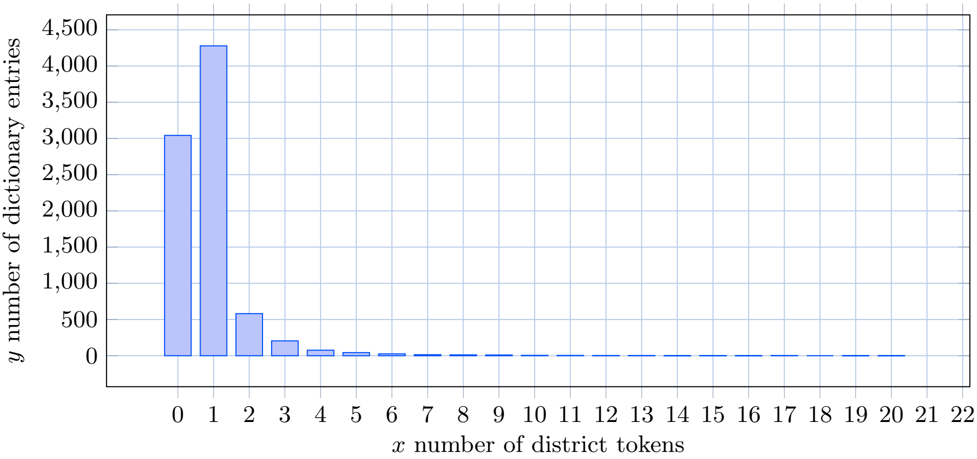


Figure 8. Number of dictionary entries per district token matches.

Phrases indicating spatial relations were recognized in parser in 24.55% of total entries, which was 38.74% of entries which recognized at least one recognized district token. Figure 9 shows how often many tokens describing spatial relations were matched by the parser. In this chart, only entries with recognized district tokens were considered. A relatively low coverage could be observed, which could be explained by the two following factors:

- spatial tokens being rare,
- parser recognizes only a subset of possible phrases.

The experiment results were shown in Table 1.

It could be observed that a steeper falloff function yields better results; however, the borderline case of the steepness – the function simply assigning the weight 1 to city closest to the capital of a district (denoted as `min`) didn't gave the best results.

The difference between using estimation with and without spatial relations was approx. 5% better for each case in favor of estimation based on spatial relations. This outcome indicates that spatial relations introduce slight improvement, and this technique is worthy of improvement in the course of further research.

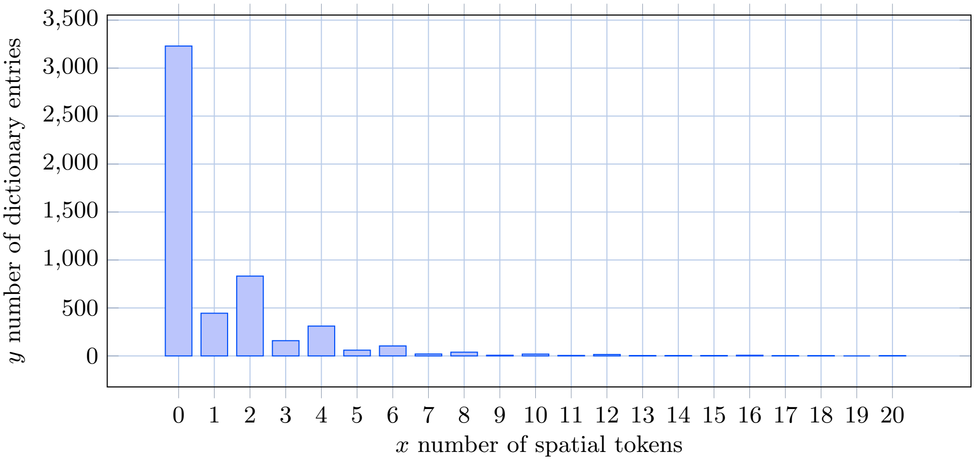


Figure 9. Number of dictionary entries per spatial token matches.

Table 1
Performance of different falloff functions.

f	average	median
$f_m \equiv 1$	126.43 km	115.87 km
$\frac{1}{1+\ln(1+d)}$	75.26 km	54.63 km
$\frac{1}{1+d^2}$	34.44 km	17.92 km
$\frac{1}{1+d^3}$	27.22 km	11.69 km
$\exp(-d^2)$	24.63 km	8.17 km
min	29.47 km	12.17 km

5. Future works

Future works will focus on improving the parser to extract more kinds of meaningful information to construct a digitized version of the dictionary. Location estimation is planned to be improved by introducing more kinds of the spatial relations between the villages. This will also improve a coverage of the dictionary text. Possibly, the algorithm could be tested against different kinds of data than what is found in the dictionary.

Acknowledgements

I would also like to thank Mrs. Ewa Wiszowata for her hard work on improving the quality of the dictionary text used in this research.

References

[1] Bajerowa I.: *Polski język ogólny XIX wieku: Składnia, synteza.* Prace naukowe Uniwersytetu Śląskiego w Katowicach. Uniwersytet Śląski, 2000.

[2] Bembenik R., Skonieczny L., Rybiński H., Niezgodka M.: *Intelligent Tools for Building a Scientific Information Platform.* Studies in Computational Intelligence. Springer, 2012.

[3] Bień J. S.: Digitalizing dictionaries of polish. In Krzysztof Bogacki, Joanna Cholewa, and Agata Rozumko, editors, *Methods of Lexical Analysis: Theoretical assumption and practical applications*, pp. 37–45. Wydawnictwo Uniwersytetu w Białymstoku, Białystok, 2009.

[4] BL Decker: World geodetic system 1984. Technical report, DTIC Document, 1986.

[5] Densham I., Reid J.: A geo-coding service encompassing a geo-parsing tool and integrated digital gazetteer service. In *Proceedings of the HLT-NAACL 2003 workshop on Analysis of geographic references — Volume 1*, HLT-NAACL-GEOREF '03, pp. 79–80, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[6] Durzewski M., Jankowski A., Szydelko L., Wiszowata E.: On digitalizing the geographical dictionary of polish kingdom published in 1880. In *Intelligent Tools for Building a Scientific Information Platform*, volume 467 of *Studies in Computational Intelligence*, pp. 53–64. Springer, 2013.

[7] Fellbaum C.: Theory and applications of ontology: Computer applications. *Media*, (2000):231–243, 2010.

[8] Gan Q., Attenberg J., Markowetz A., Suel T.: Analysis of geographic queries in a search engine log. In *Proceedings of the first international workshop on Location and the web*, LOCWEB '08, pp. 49–56, New York, NY, USA, 2008. ACM.

[9] Gey F., Larson R., Sanderson M., Joho H., Clough P., Petras V.: GeoCLEF: The CLEF 2005 Cross-Language Geographic Information Retrieval Track Overview. pp. 908–919. 2006.

[10] Grover C., Tobin R., Byrne K., Woollard M., Reid J., Dunn S., Ball J.: Use of the Edinburgh geoparser for georeferencing digitized historical collections. *Philosophical Transactions of The Royal Society A: Mathematical, Physical and Engineering Sciences*, 368:3875–3889, 2010.

[11] Kaplan D. H.: *Boundaries and place: European borderlands in geographical context.* Rowman & Littlefield Publishers, 2002.

[12] Lévy A.: *Basic Set Theory.* Number v. 13 in Dover Books on Mathematics Series. Dover, 2002.

[13] Nilsson U., Małuszyński J.: *Logic, programming, and Prolog.* Wiley, 1990.

[14] Rumbaugh J., Blaha M., Premerlani W., Eddy F., Lorenson W.: *Object-Oriented Modeling and Design.* Prentice Hall, Inc., 1^{st} edition, October 1991.

[15] Russell S. J., Norvig P., Candy J. F., Malik J. M., Edwards D. D.: *Artificial intelligence: a modern approach.* Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1996.

[16] Schantz H. F.: *The history of OCR, optical character recognition.* Recognition Technologies Users Association, 1982.

[17] Stubblebine T.: *Regular Expression Pocket Reference: Regular Expressions for Perl, Ruby, PHP, Python, C, Java and. NET.* O'Reilly Media, Incorporated, 2007.

[18] Sulimierski F., Chlebowski B., Walewski W.: *Słownik geograficzny Królestwa Polskiego i innych krajów słowiańskich.* Number v. 1–15 in Słownik geograficzny Królestwa Polskiego i innych krajów słowiańskich. Wydawnictwa Artsytyczne i Filmowe, 1902.

[19] Wieczorek J., Guo Q., Hijmans R.: The point-radius method for georeferencing locality descriptions and calculating associated uncertainty. *International Journal of Geographical Information Science*, 18(8):745–767, 2004.

Affiliations

Grzegorz Jaśkiewicz
Warsaw University of Technology – The Faculty of Electronics and Information Technology,
ul. Nowowiejska 15/19, 00-665 Warsaw, e-mail: `grzegorz@jaskiewi.cz`

Received: 8.02.2013
Revised: 28.04.2013
Accepted: 28.04.2013