Experiments with ad hoc ambiguous abbreviation expansion

Agnieszka Mykowiecka
ICS PAS
Jana Kazimierza 5
Warsaw, Poland
agn@ipipan.waw.pl

Małgorzata Marciniak
ICS PAS
Jana Kazimierza 5
Warsaw, Poland
mm@ipipan.waw.pl

Abstract

The paper addresses experiments to expand ad hoc ambiguous abbreviations in medical notes on the basis of morphologically annotated texts, without using additional domain resources. We work on Polish data but the described approaches can be used for other languages too. We test two methods to select candidates for word abbreviation expansions. The first one automatically selects all words in text which might be an expansion of an abbreviation according to the language rules. The second method uses clustering of abbreviation occurrences to select representative elements which are manually annotated to determine lists of potential expansions. We then train a classifier to assign expansions to abbreviations based on three training sets: automatically obtained, consisting of manual annotation, and concatenation of the two previous ones. The results obtained for the manually annotated training data significantly outperform automatically obtained training data. Adding the automatically obtained training data to the manually annotated data improves the results, in particular for less frequent abbreviations. In this context the proposed a priori data driven selection of possible extensions turned out to be crucial.

1 Introduction

Saving time and effort is a crucial reason for using abbreviations and acronyms in all types of texts. In informal texts like e-mails, communicator messages, and notes, it is very common to create ad hoc abbreviations, which are easy for the author (in the case of personal notes) or a reader to interpret in the context of a topic discussed by a group of people.

The time/effort saving principle is also valid for medical notes prepared by physicians during patient visits, hospital examinations, and for nursing notes. They are often written in a hurry, but have to be understandable for other people involved in the treatment of a patient. They cannot be completely hermetic, but usually, they are difficult to interpret both for patients and nonspecialists (Mowery et al., 2016b). Ad hoc abbreviations are also difficult for automatic data processing systems to handle. But proper understanding and normalization of all types of abbreviations is indispensable for the correct operation of information extraction systems (Pradhan et al., 2014), data classification (Névéol et al., 2016; Mowery et al., 2016a), question answering (Kakadiaris et al., 2018), and many other applications.

The interpretation of an abbreviation consists of two aspects: its recognition and expansion. The recognition of well established abbreviations and acronyms is usually done with the help of dictionaries. For the English medical domain, several dictionaries such as the resources of the U.S. National Library of Medicine are available. Ad hoc abbreviations are not present in dictionaries and they are mainly recognized as unknown words. Sometimes, they are ambiguous with correct full word forms listed in general language dictionaries. For example, *dept* might be an abbreviation of 'department' or 'deputy'; in Polish medical notes *temp* 'rate' is an abbreviation of *temperatura* 'temperature'. In informal texts, the period after abbreviations, required after some of them in Polish, English and many other languages, is often omitted, which makes the distinction between word and abbreviation more difficult. Ad hoc abbreviations are also ambiguous with standard language abbreviations, and their interpretation is different from those used in standard language: literature, papers or everyday use. For example, in many languages (e.g. English, German, Polish) the abbreviation *op* means the opus number in musical composition. In Polish medical notes, it can be *opa-*

Proceedings of the 10th International Workshop on Health Text Mining and Information Analysis (LOUHI 2019), pages 44–53
Hong Kong, November 3, 2019.
https://doi.org/10.18653/v1/D19-62

trunek 'dressing' especially in the context of 'gypsum dressing' of broken bones, *oko prawe* 'right eye' in the context of ophthalmic examinations or *opakowanie* 'package' of medications in recommendations. But it can have several other meanings e.g. *opuszek* 'fingertip' – unit of cervical dilation used by gynecologists. So, if we want to recognize ad hoc abbreviations in informal texts it is necessary not only to consider unknown strings but also short words and abbreviations recognized during morphologic analysis of text.

Our study focuses on expanding word abbreviations in medical notes in Polish. We are not systematically considering phrase abbreviations, usually called acronyms, as selecting candidates for their expansions requires different methods. In the paper we test two approaches for selecting candidates for word abbreviation expansions which are used for training a classifier to assign the appropriate expansion to an abbreviation, see (Pakhomov, 2002). In the first method, we check a hypothesis that full forms of ad hoc abbreviations are represented in texts of the same domain and type. So, for each candidate for abbreviation, we select all words which might be expansions for the abbreviation according to the language rules. We test if the occurrences of potential expansions in text can be sufficient for training the classifier. This method provides us with many suggestions which might never be used. To limit this number, we modified the method by selecting words with distributed representation close to the representation of the abbreviation. In the second method, we select candidates for abbreviation expansions based on annotation of selected elements of abbreviation occurrences clusters. Clustering is done by the Chinese whispers algorithm (Biemann, 2006) according to their contexts. For each cluster, we expand a manually randomly selected 2 to 6 elements of each cluster. This procedure gives us a short list of potential expansions.

2 Related Work

The problem of abbreviation recognition and expansion, has so far been addressed mainly for English data, e.g. (Nadeau and Turney, 2005) and (Moon et al., 2012) where supervised machine learning algorithms are used, and (Du et al., 2019) who describes a complex system for English data that recognizes many types of abbreviations. But, there are papers describing the problem for other languages too, e.g. Swedish (Kvist and Velupillai, 2014) – SCAN system based on lexicon, and German, e.g. (Kreuzthaler et al., 2016) where abbreviation identification linked to the disambiguation of period characters in clinical narratives is addressed.

Methods of dealing with acronyms are described among others in (Schwartz and Hearst, 2003) where the authors look for acronym definitions in data and identify them as a text in parentheses adjacent to the acronym/abbreviation; (Tengstrand et al., 2014) where experiments for Swedish are described; and (Spasic, 2018) where terminology extraction methods are applied.

Experiments in which similar to our data driven approach is tested, are described in (Oleynik et al., 2017). They used a method of abbreviation expansion based on N-grams and achieved an F1 score of 0.91, evaluated on a test set of 200 text excerpts.

3 Data

Medical reports which we used to carry out the experiments are an annomyzed sample of data collected by the company providing electronic health record services. The research is a part of a project, the purpose of which is, among other things, automatic preparation of statistics concerning data on diseases, symptoms and treatments. The statistics should be based on information extracted automatically from descriptions of patients visits. The data was collected in many clinics and concerned visits to doctors of various specialties. Identification information was removed from texts, and only descriptions of interrogation, examination and recommendations were processed.

As Polish is an inflectional language we preprocessed text to obtain base word forms and POS tags. Medical reports usually contain a limited dictionary but a lot of words are not present in general dictionaries, thus specialized medical taggers would be the most appropriate for performing this task. However, manually annotated data to train the medical tagger is not available for Polish, thus we had to processed texts with the general purpose morphological tagger. In this work we used Concraft2 – the new version of the Concraft (Waszczuk, 2012) tagger which cooperates with the general purpose morphological analyzer Morfeusz2 (Woliński, 2014) and also performs tokenization and sentence identification. Additionally, we ensured that line breaks were treated

as sentence delimiters, as often a dot was not used at the end of a line, while the topic was changed. The quality of automatic tagging of medical texts in Polish is not high, see (Marciniak and Mykowiecka, 2011). Medical notes contain a large number of spelling errors, there are many acronyms/abbreviations[1] and specialized words not present in the general purpose morphological analyzer. Thus, we performed our experiments using both exact word forms and lemmas.

The entire dataset consists of about 10 million tokens and 15,000 different word forms. This number is larger than for English data of the same size as Polish is an inflectional language. It means that one word can have several forms, e.g.: *kropla* 'drop' is represented in our data as: *krople*, *kropli*, *kroplach*, *kroplami*. As we differentiate between capital and small letters we additionally have the following forms: *Krople*, *KROPLE*. The latter decision resulted from a desire to preserve information about acronyms for future work. For example, the form *PSA* is rather the acronym of an examination while the form *psa* might be interpreted as a 'dog' in the genitive (in medical texts it mainly occurs in the phrase *sierść psa* 'dog fur' in the context of allergens).

Around 7% of tokens are recognized as unknown words and this group of tokens consists of about 91,000 different elements: abbreviations; acronyms; proper names such as medications and illnesses containing proper names (e.g. Hashimoto's thyroiditis); and typos that occur in large numbers in medical notes. Some abbreviations are represented in Morfeusz2 but often their meaning is not appropriate for medical texts, e.g.: a string 'por' is recognized as an abbreviation of 'lieutenant' or 'compare' while in medical data it is 'clinic'.

Tokens which are not recognized by dictionaries, are natural candidates for being abbreviations. In many papers addressing the problem of abbreviation recognition, the authors limit themselves to considering such tokens, see (Park and Byrd, 2001), (Kreuzthaler et al., 2016). In our approach, when selecting potential abbreviations, we took into account all forms out of the dictionary, and short words (up to 5 letters) which were in the dictionary. As we wanted to use contexts in our experiment, we decided to consider forms which occurred in the data more than 15 times. This limited the list of unknown tokens to 2808 and the list of word forms considered as potential abbreviations to 3152.

The data set was divided into ten parts, one was left for evaluation purposes and the remaining 9 were used as a training set and a source of information on the number and types of abbreviations used.

The test set consists of about 996.000 tokens and thousands of abbreviation occurrences. To make manual checking of the results feasible we decided to perform our experiment on a small subset of 15 abbreviations. This short list consists of abbreviations which seem to be ambiguous (a few likely interpretations) and are rather common – 3069 occurrences in test data which means 0.3% of tokens. Their proper recognition is, therefore, important for correct text interpretation. All occurrences of these 15 abbreviations in the test set were manually expanded by a person with experience in medical data processing. All difficult cases were consulted with a specialist. A fragment of the list together with exemplary variations is given in Table 1.

4 Language Models

On the basis of the entire data set, we trained four word2vec (Mikolov et al., 2013) versions of language models (the choice of specialized data seems to be straightforward, but was also supported by (Charbonnier and Wartena, 2018)). One pair of models was trained on the original (tokenized) texts – inflected forms of words. The second pair of models was trained on the lemmatized text (in Polish, nouns, verbs and adjectives have many different inflectional variants). In both pairs we calculated vectors of length 100; one model was trained on all tokens which occurred at least 5 times and the second one was trained on text in which all numbers were replaced by one symbol. In the final experiments form based models turned out to be the most efficient.

5 Baseline

We solve the problem of abbreviation expansion as the task of word sense disambiguation where a classifier is trained on all expansions represented in the data. As it is difficult to compare our approach to other work as the assumptions of the tasks related to abbreviation expansion were dif-

[1] Marciniak and Mykowiecka (2011) reported that around 6% of tokens in hospital records are acronyms and abbreviations.

Abr.	Variations	All	Possible Meanings in Test Data
fiz	FIZ, fiz, Fiz,	15	fizjologiczny, fizycznie, fizyczny, fizykalnie, fizykalny, fizykoterapia *Physiological, Physically, Physical, Physical, Physical Therapy*
cz	CZ, Cz	44	czerwień, czołowy, czynnik, czynność, częstość, część *redness, frontal, factor, activity, frequency, part*
gł	gł	22	głowa, główkowy, głównie, główny, głęboki *head, head(adj), mainly, main, deep*
op	OP Op	30	opak, opakowanie, opatrunek, opera, operacja, operacyjnie, operacyjny, oko prawy operowany, opieka, opis, opuszek, opór *awry, package, dressing, opera, operation, operationally, operational, right eye operated, care, description, pad, resistance*

Table 1: Four from the list of 15 abbreviations with variations, the number of all different longer words found in the training data.

	simulated train data			annotated	
abbr.	AL	SL	CL	train	test
cz	51022	46172	25642	96	137
fiz	10769	10684	9895	61	59
gł	15591	14460	9988	55	48
kr	37381	24349	20053	81	224
mies	9021	8949	6874	35	206
op	24677	21673	9285	410	1785
poj	4386	4035	3293	75	147
pow	22517	5037	17271	69	65
pr	88312	20386	57809	105	100
rodz	6459	6459	4903	26	52
śr	3894	2922	2316	61	65
wz	9942	6914	3345	42	31
zab	8670	8085	7755	69	90
zaw	3826	1296	2012	28	29
zał	1657	1544	717	18	31
total	298149	182965	181140	1231	3069

Table 2: Number of occurrences in train and test data. The three potential extensions lists for simulated training sets: AL – all words being potential expansions, SL – all the possible words in our distributional model whose similarity to a particular abbreviation was higher than 0.2 for a language model created on forms, CL – annotations of randomly selected cluster elements.

ferent, we suggest an artificial baseline, which consists of the most common interpretation of manually annotated abbreviations in the test set. Table 3 gives appropriate statistics. If we assign the most common interpretation of an abbreviation to all its occurrences we obtain the weighted precision equal to 0.568, the recall equal to 0.742 and the F1 measure equal to 0.64.

6 Methods for Determining Expansions

We checked two methods for determining potential ad hoc abbreviation expansions. The first one assumes that full versions of abbreviated forms are available somewhere in the data. So the problem can be seen as an attempt to determine which words from the text data can be abbreviated to a considered token and which of them correspond to an abbreviation in the given context. The second method uses clustering of abbreviation occurrences to select representative elements from each cluster to determine lists of potential expansions. This method allows a considered token that can abbreviate a phrase to be taken into account, while the first method is only oriented on word expansions.

6.1 All Words and Similar Word Methods

When we look for potential expansions of a selected token without any additional resources, we have to consider two cases. The first, is that we should leave the token unchanged as it could be a correct word or acronym. We do not address this problem. The second, is that we should select all words from the data that can be abbreviated to the considered token according to language rules. So, the list of potential expansions consists of all forms from the data which met the conditions of being an abbreviation in Polish. We analyse cases in which a token x might be an abbreviation of a word y if:

- the beginning of y is equal to x;
- the POS tag does not indicate an abbreviation or an unknown word (to avoid using incorrectly written words as potential extensions);
- the abbreviation does not cross Polish two-letter compounds (‘rz’, ‘sz’, ‘cz’, ‘ch’).

The first potential extensions list (AL) contains all words meeting the above conditions. It consists of 1345 elements. The AL list contains forms which are never shortened. Their usage should thus be different from that of the abbreviation itself. To eliminate such unlikely expansions and to limit the number of potential labels, we selected

abbr.	test anot.	expansions
cz	137	czerwień(1), czynnik(3), czynność(14), częstość(14) część**(102)** *redness, factor, activity, frequency, part*
fiz	59	fizjologiczny(2), fizycznie(1), fizyczny(5),fizykalnnie**(45)**, fizykalny(6) *physiological, physically, physical, physically, physical*
gł	48	gładki(1), głowa(16), główkowy(4), głównie**(17)**, główny(10) *smooth, head, head, mainly, main*
kr	224	krawędź(1), kreatynina(2), kropla(68), krople(4), kręgosłup**(149)** *edge, creatinine, drop, drops, spine*
mies	206	miesiąc**(187)**, miesiączka(18), miesięczny(1) *month, menstruation, monthly*
op	1785	oko prawe (349), ostatni poród (1), opakowanie**(1384)**, opatrunek(6), operacja(22), operacyjnie(3), operacyjny(10), operować(1), opieka(7), opuszek(2) *right eye, last delivery, package, dressing, surgery, surgically, surgical, operate, care, fingertip*
poj	147	pojawić(2), pojedynczy**(127)**, pojemnik(18) *appear, single, container*
pow	65	powierzchnia(17), powiększony(6), powiększyć(8), powlekać(9), powyżej**(24)**, powód(1) *surface, enlarged, enlarge,coated, above, reason*
pr	100	Pogotowie Ratunkowe(5), public relations(3), prawa ręka(1), PR(1), per rectum(5), prawidłowo(17), prawidłowy**(34)**, prawy(20), preparat(2), prostata(4), przewód(1), przychodnia(1), próba(6) *Emergency Service, public relations, right hand, PR(in ECG), per rectum, properly, normal, right, preparation, prostate, tract, clinic, test*
rodz	52	rodzeństwo(8), rodzice(3), rodzina(4), rodzinne(1), rodzinnie**(20)**, rodzinny(16) *sibling, parents, family, family, family, family*
wz	31	wziernik**(9)**, wzrost(4) *speculum, high*
śr	65	średni(3), średnica**(47)**, średnio(10), środa(1), środek(3), środkowy(1) *medium, diameter, medium, Wednesday, middle, middle*
zab	90	zabieg(14), zaburzenie**(76)** *surgery, disorder*
zaw	29	zawiesina**(23)**, zawód(6) *suspension, profession*
zał	31	załamek(10), założyć(5), załączyć**(16)** *crinkle, put on, attach*

Table 3: Test set abbreviation expansions in numbers

from all the possible word forms in our distributional model, those whose similarity to a particular abbreviation was higher than 0.2 for a language model created on forms, see Section (4). These candidates form the second expansion list of 259 elements (SL). The numbers of occurrences of all expansions of these three lists in the training data are given in Table 2.

6.2 Clustering and Manual Annotation

To check whether abbreviation usages form any differentiable clusters, we identified all their occurrences in the training data. For each such occurrence, we determined the context vector, which was equal to the average of vectors of surrounding tokens. In the experiment, we set the context as three tokens before and after each abbreviation. Then, we clustered occurrences of the abbreviation via the Chinese whispers algorithm (Biemann, 2006) which does not impose defining a priori a number of clusters. As we aimed to select examples of various interpretations of the same abbreviation and various usage of the same interpretation of the abbreviation, we established quite a high level of similarity between nodes in the initial graph. The similarity was counted as the cosine between vectors and we set it experimentally to 0.7 (it had no theoretical justification). Increasing the parameter of similarity we obtain more clusters and they represent higher granularity of abbreviation contexts.

For each cluster, we randomly selected from 2 to 6 elements (depending on the cluster size) and manually annotated them and the representative elements of the cluster pointed out by the algorithm, with proper expansions in the data. 85 elements used in this manual annotation constitutes the third list (CL) used in our experiments. In Table 2 the number of annotated examples in both train and test data are give. In test data a very high variance of abbreviations occurrences caused mainly by the big number of clusters obtained for the most frequent abbreviation (*op*) can be seen (from 29 to 1785).

6.3 Training Data

The core training set (SIM) is constructed via simulation by shortening word forms beginning with any of the abbreviation from the appropriate list processed in the exact experiment: AL, SL, and CL. The longest list AL contains 1345 potential expansions, SL limits the number of potential expansions to 259 elements while the manually created list CL has 85 elements. However, the SIM set may be biased as some of the words from these lists might be never shortened. What is more, in some typical places in which the chosen abbreviations occur, the full form may never or almost never be used. To check the real value of such simulated training set and, to test if a much smaller training set could be sufficient for this task, we also prepared manually annotated training set. It was built from all the manually annotated examples (a procedure described in 6.2). In Table 2, there is a comparison of the numbers of considered abbreviations in simulated (depending on the chosen expansion list) and manually annotated training and test data. As it turned out that nearly every abbreviation can also be an acronym, and one of them *oko prawy – op* occurs many times in our annotated data, to make comparisons more complete we also prepared a version of our training data (SIM-ac) in which two consecutive words recognized during manual annotation as a possible full form of an acronym, are abbreviated to the sequence of their first letters.

7 Neural Net Architecture

In the experiment, we used bidirectional LSTM nets as being most frequently judged as good for sequence processing. We formulated our task as a prediction task in which we predict a word on the basis of its context (and, optionally, on the basis of a representation of the abbreviation used instead of it). As clinical notes are short, concise and frequently change subject, we assign a label (which is a full word form) to a word on the basis of its left and right contexts of 3 or 5 words.

Input to a net consists of a subset of the following data (names given after features descriptions are used in Table 4 headings):

- word vectors form the models trained on the entire dataset,
- POS tags encoded as one-hot vector (pos) (31 most frequent categories),

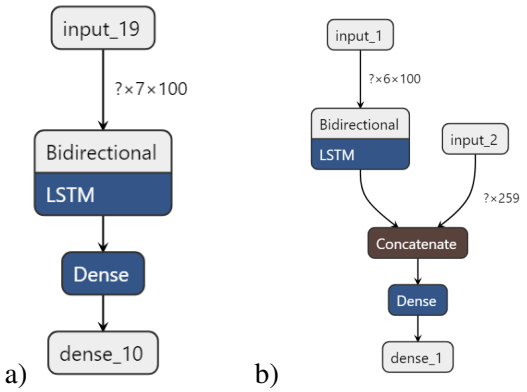

Figure 1: Two neural net architectures tested

- vector representing an abbreviation itself (padded with zeros if needed), (c),
- all possible longer versions of the particular abbreviations coded in a vector representing all possible forms of all the abbreviations taken into account. This vector was added as the additional information which was combined with the LSTM output layer (only one output value is considered), (cd), Figure 1b.

Two net architectures tested are given in Figure 1.[2] On the left side, a basic BiLSTM net with input consisting of seven word representation is shown (the central word is an abbreviation – being actually in the data or inserted in place of a full word form in the simulation variant). Representation consisted either of word embeddings only, or of embeddings concatenated with the POS one-hot encodings. We also tested variants in which only context words were used. The architecture given on the right takes only context words as input. The additional input vector represents all valid extensions of a given abbreviation (cd). In both cases, the last layer is a standard classification dense layer with a sigmoid activation function. Its size depends on the length of a particular extensions list. The implementation is done in Keras with Tensorflow backend. The Adam optimizer was used and the other settings are standard values used in Keras implementation.

8 Results

The net architecture for further experiments was chosen on the basis of the 10-fold cross validation results for one configuration, see Table 4.

[2]The picture was obtained using Netron https://lutzroeder.github.io/netron/

The number of epochs was established on the basis of validation on 1/10th of the training data while learning on the remaining training set to 2 for models which use big simulated data, and 5 for models which use only small annotated data. The batch size was equal to 32 and 1 respectively. Apart from the first model which does not take into account either the abbreviation representation (c) or the list of possible expansions (cd), other results do not vary much. We decided to choose the second best variant with the list of possible extensions added as an additional layer but with the three not five word context. As our annotated data is not big, we preferred a comparable model with fewer parameters.

The chosen set of features (second column in Table 4) was used for building models for all abbreviation lists and four variants of training sets (only annotated data – ANOT, only simulated – SIM (only word abbreviations) and SIM-ac (word abbreviations and acronyms), and the sum of both.

The results on cross validation (Table 5) are the best for simulated data, probably because of their size and repetitiveness. Only for the longest expansion list the results are the best on the smallest annotated set. The small number of examples could have just been memorized more easily. With this one exception it is also generally the rule that the bigger the training set the better, although adding annotated examples lowered the results slightly for SL list.

The results for the test set (Table 6) are better than those obtained for the cross validation on the training set in many cases. This is probably due to the small size of the test data set and many occurrences of the easy to resolve cases, for example the frequent occurrences of the 'op' abbreviation, which was correctly identified as 'opakowanie' *package*. However, models trained on the simulated set alone, performed significantly worse in terms of the recall (precision only deteriorated a little).

Using simulated data for training models has one important advantage – it saves time and effort. But there are also some disadvantages which have to be carefully analyzed. A few examples of miss-interpretation of 'pr' strings are given in Table 7. In our particular task, the possible problems can have different sources. First, some abbreviations are never (or almost never) expanded within the corpus. These are for example very common acronyms (like *OP* – 'right eye') which are rarely written in the full form, or an abbreviation *meta* which is never used in its full form *metastaza* 'metastasis' in our corpus. We did not fully address the problem in this work and phrase extensions which were recognized during manual annotation were added manually to the expansion lists. The second problem is that some words are never abbreviated, but we automatically added them to our expansion lists making the problem harder to solve. However, the good results obtained for the AL list show that this situation was not very confusing for our models (which have access to annotated data). The third problem is the fact that the contexts in which the abbreviated form are used may differ from the contexts where the full form occurs. If in some contexts, only abbreviations are used and the full form never occurs, it is not possible to learn this pattern. For example, when prescribing the number of medicine packages, doctors always use *op* instead of *opakowanie* 'package', e.g. *Lantus (1 op. 30%)*. Our experiment confirmed that this is really the case. Results obtained by the models trained on simulated data only, although having very good cross validation results, have much worse recall on test data than models trained on annotated examples. However, adding annotated data to the simulated train set improved the results. For all but the AL list, the results obtained on the entire data even outperformed those obtained for the annotated data.

9 Conclusions

In the paper, we wanted to test if simulated abbreviations can be used to expand ambiguous ad hoc abbreviations in medical notes. Although simulation of the training data is a very useful practice, as manual data annotation is an expensive and time consuming process, our work shows that the obtained results are not always satisfactory. The F1 measure we obtained is below the artificially established baseline (the F1 measure equal to 0.64). Moreover, the experiments show that annotation of a small number of thoroughly selected examples of abbreviation occurrences gives satisfactory results for the task with the F1 measure equal to 0.92. It significantly outperforms the artificial baseline – the most common expansion e.g. the standard baseline for the word sense disambiguation task. However, the best results are obtained when the simulated data are combined with man-

	context=3, model based on forms					lemas	context=5
	pos-c-cd-	**pos-c-cd+**	pos-c+cd-	pos-cd+c+	pos+c-cd+	pos-c-cd+	pos-c-cd+
weighted precision	0.562	0.690	0.662	0.691	0.689	0.677	0.693
recall	0.652	0.770	0.757	0.764	0.764	0.764	0.778
F1	0.597	**0.721**	0.702	0.719	0.717	0.711	**0.726**
macro precision	0.367	0.469	0.458	0.471	0.472	0.475	0.484
recall	0.391	0.502	0.493	0.493	0.505	0.492	0.519
F1	0.368	0.476	0.465	0.472	0.478	0.473	0.491

Table 4: Results for 10-fold cross validation for different bidirectional LSTM settings for one training set (a subset of randomly selected cluster elements) and a chosen extension list. In all but the sixth case, word embeddings based on word forms were used. Additional information used in the models: pos – part of speech, c – vector representing an abbreviation itself, cd – vector codding possible extensions of the particular abbreviations (architecture from Figure 1b).

List / Trainset	AL			SL			CL		
	P	R	F1	P	R	F1	P	R	F1
ANOT-rd	0.874	0.885	**0.875**	0.809	0.837	0.817	0.866	0.888	0.875
ANOT	0.854	0.869	0.853	0.855	0.855	0.855	0.884	0.901	0.891
SIM	0.864	0.872	0.866	0.896	0.901	**0.897**	0.968	0.969	**0.968**
ANOT+SIM	0.864	0.872	0.866	0.893	0.899	0.894	0.968	0.969	**0.968**

Table 5: Results for 10-fold cross validation of the selected net architecture for all extension lists and training set variants (notation explained in the text). The three potential extensions lists for simulated training sets: AL – all words being potential expansions, SL – all the possible words in our distributional model whose similarity to a particular abbreviation was higher than 0.2 for a language model created on forms, CL – annotations of randomly selected cluster elements. The best results for each expansion list are shown in bold.

List / Model trained on	AL			SL			CL		
	P	R	F1	P	R	F1	P	R	F1
	weighted results								
ANOT	0.914	**0.926**	**0.917**	0.891	**0.906**	**0.893**	0.909	0.921	0.910
SIM	0.800	0.482	0.556	0.770	0.545	0.611	0.806	0.735	0.758
SIM-ac	0.907	0.386	0.441	0.888	0.472	0.537	0.930	0.748	0.777
ANOT+SIM	**0.947**	0.749	0.824	0.911	0.749	0.809	**0.944**	0.911	0.918
ANOT+SIM-ac	**0.947**	0.715	0.798	**0.915**	0.776	0.828	0.943	**0.928**	**0.930**
	macro results								
ANOT	0.516	0.514	0.500	0.492	0.513	0.479	0.495	0.540	0.489
SIM	0.308	0.314	0.286	0.317	0.359	0.304	0.460	0.539	0.469
SIM-ac	0.302	0.299	0.268	0.329	0.354	0.309	0.499	0.522	0.461
ANOT+SIM	0.357	0.363	0.343	0.372	0.409	0.370	0.571	0.616	0.570
ANOT+SIM-ac	0.384	0.383	0.369	0.366	0.406	0.366	0.546	0.588	0.546

Table 6: Results for the test set of the models trained on different datasets for all extension lists (notation explained in the text). The best results for each expansion list are shown in bold. The **artificial baseline results**, when we consider only those expansions which really occurred in the data and the most frequent expansion is taken as a solution, are (weighted) **P=0.568, R=0.742, F1=0.64**. Most of our results are well above this baseline and only models trained on simulated data gave lower results on two expansions lists.

ual annotation. Is it particularly important for less frequent expansions, as the increase of macro F1 is significantly greater than increase of the weighted one. This conclusion is somewhat in contradictions with a claim of (Oleynik et al., 2017) who suggested that the manual annotation is not necessary to obtain relatively high results. In this context, the suggested method of selecting extensions candidates turned out to be important – the results on the list of every possible word extension (the AL list) for the combined training set are much lower than for the SL and CL lists.

As the results obtained for the SL expansion list (a list of all words from the data whose dis-

Excerpt	Expansion	SIM	ANOT+SIM
Jama ustna, gardło: pr [line break] 'Mouth,throat: normal'	*prawidłowy* 'normal'	*prawy* 'right'	*prawidłowy* 'normal'
bez o. patologicznych, pr. Romberg [aprawidlowa] 'without pathological symptoms, Romberg's test [spelling error]	*próba* 'test'	*prawidłowy* 'normal'	*prawidłowy* 'normal'
ogr. ruchomości kolana pr, przykurcz 'limitation of the right knee mobility, contracture'	*prawy* 'right'	*prawy* 'right'	*prawidłowy* 'normal'

Table 7: Examples of miss-interpretation of 'pr' for the CL list of potential expansions and for two training data: SIM and ANOT+SIM.

tributional similarity was higher than 0.2) and the ANOT+SIM training data are very good, it would be interesting to test how important the selection of annotated examples is and to test how many manually annotated data is necessary for obtaining satisfactory results. In the future work we want to test our method on a large set of abbreviations and include strings which are ambiguous between words and abbreviations.

Acknowledgments

This work was supported by the Polish National Science Centre project 2014/15/B/ST6/05186 and by EU structural funds as part of the Smart Growth Operational Programme POIR.01.01.01-00-0328/17

References

Chris Biemann. 2006. Chinese whispers: an efficient graph clustering algorithm and its application to natural language processing problems. In *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing*, TextGraphs-1, pages 73–80. Association for Computational Linguistics.

Jean Charbonnier and Christian Wartena. 2018. Using-word embeddings for unsupervised acronym disambiguation. In *Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, New Mexico, USA*, pages 2610—-2619.

Xiaokun Du, Rongbo Zhu, Yanhong Li, and Ashiq Anjum. 2019. Language model-based automatic prefix abbreviation expansion method for biomedical big data analysis. *Future Generation Computer Systems*, (98):238–251.

Ioannis A. Kakadiaris, George Paliouras, and Anastasia Krithara. 2018. *Proceedings of the 6th BioASQ Workshop A challenge on large-scale biomedical semantic indexing and question answering*. Association for Computational Linguistics, Brussels, Belgium.

Markus Kreuzthaler, Michel Oleynik, Alexander Avian, and Stefan Schulz. 2016. Unsupervised abbreviation detection in clinical narratives. In *Proceedings of the Clinical Natural Language Processing Workshop (ClinicalNLP)*, pages 91–98. The COLING 2016 Organizing Committee.

Maria Kvist and Sumithra Velupillai. 2014. SCAN: A swedish clinical abbreviation normalizer - further development and adaptation to radiology. In *Information Access Evaluation. Multilinguality, Multimodality, and Interaction - 5th International Conference of the CLEF Initiative, CLEF 2014, Sheffield, UK, September 15-18, 2014. Proceedings*, volume 8685 of *Lecture Notes in Computer Science*, pages 62–73. Springer.

Małgorzata Marciniak and Agnieszka Mykowiecka. 2011. Towards Morphologically Annotated Corpus of Hospital Discharge Reports in Polish. In *Proceedings of BioNLP 2011*, pages 92–100.

Thomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Sungrim Moon, Serguei Pakhomov, and Genevieve B. Melton. 2012. Automated disambiguation of acronyms and abbreviations in clinical texts: window and training size considerations. In *AMIA Annu Symp Proc.*, pages 1310–9.

Danielle. Mowery, Brian Chapman, Mike Conway, Brett South, Erin Madden, Salomeh Keyhani, and Wendy Chapman. 2016a. Extracting a stroke phenotype risk factor from veteran health administration clinical reports: An information content analysis. *Journal of Biomedical Semantics*, 7(1).

Danielle L. Mowery, Brett R. South, Lee Christensen, Jianwei Leng, Laura-Maria Peltonen, Sanna Salanterä, Hanna Suominen, David Martinez, Sumithra Velupillai, Noémie Elhadad, Guergana Savova, Sameer Pradhan, and Wendy W. Chapman. 2016b. Normalizing acronyms and abbreviations to aid patient understanding of clinical texts: Share/clef ehealth challenge 2013, task 2. *Journal of Biomedical Semantics*, 7(1).

David Nadeau and Peter D. Turney. 2005. A supervised learning approach to acronym identification. In *Proceedings of the 18th Canadian Society Conference on Advances in Artificial Intelligence*, AI'05, pages 319–329, Berlin, Heidelberg. Springer-Verlag.

Aurélie Névéol, Cyril Grouin, Kevin B Cohen, Thierry Hamon, Thomas Lavergne, Liadh Kelly, Lorraine Goeuriot, Grégoire Rey, Aude Robert, Xavier Tannier, and Pierre Zweigenbaum. 2016. Clinical information extraction at the CLEF eHealth evaluation lab 2016. In *Proc of CLEF eHealth Evaluation lab*, pages 28–42, Evora, Portugal.

Michel Oleynik, Markus Kreuzthaler, and Stefan Schulz. 2017. Unsupervised abbreviation expansion in clinical narratives. In *MedInfo*, volume 245 of *Studies in Health Technology and Informatics*, pages 539–543. IOS Press.

Serguei Pakhomov. 2002. Semi-supervised maximum entropy based approach to acronym and abbreviation normalization in medical texts. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 160–167, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Youngja Park and Roy J. Byrd. 2001. Hybrid text mining for finding abbreviations and their definitions. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*.

Sameer Pradhan, Noémie Elhadad, Brett R South, David Martinez, Lee Christensen, Amy Vogel, Hanna Suominen, Wendy W Chapman, and Guergana Savova. 2014. Evaluating the state of the art in disorder recognition and normalization of the clinical narrative. *Journal of the American Medical Informatics Association*, 22(1):143–154.

Ariel S. Schwartz and Marti A. Hearst. 2003. A simple algorithm for identifying abbreviation definitions in biomedical text. In *Pacific Symposium on Biocomputing*, pages 451–462.

Irena Spasic. 2018. Acronyms as an integral part of multi-word term recognition - A token of appreciation. *IEEE Access*, 6:8351–8363.

Lisa Tengstrand, Beáta Megyesi, Aron Henriksson, Martin Duneld, and Maria Kvist. 2014. EACL - expansion of abbreviations in clinical text. In *PITR@EACL*, pages 94–103. Association for Computational Linguistics.

Jakub Waszczuk. 2012. Harnessing the CRF complexity with domain-specific constraints. the case of morphosyntactic tagging of a highly inflected language. In *In: Proceedings of COLING 2012, Mumbai, India.*

Marcin Woliński. 2014. Morfeusz reloaded. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014*, pages 1106–1111, Reykjavík, Iceland. ELRA.