Lang Resources & Evaluation (2014) 48:227–248
DOI 10.1007/s10579-013-9246-z

ORIGINAL PAPER

General framework for mining, processing and storing large amounts of electronic texts for language modeling purposes

Jan Švec · Jan Lehečka · Pavel Ircing · Lucie Skorkovská · Aleš Pražák · Jan Vavruška · Petr Stanislav · Jan Hoidekr

Published online: 24 July 2013

Abstract The paper describes a general framework for mining large amounts of text data from a defined set of Web pages. The acquired data are meant to constitute a corpus for training robust and reliable language models and thus the framework needs to also incorporate algorithms for appropriate text processing and duplicity detection in order to secure quality and consistency of the data. As we expect the resulting corpus to be very large, we have also implemented topic detection algorithms that allow us to automatically select subcorpora for domain-specific language models. The description of the framework architecture and the implemented algorithms is complemented with a detailed evaluation section. It analyses the basic properties of the gathered Czech corpus containing more than one billion text tokens collected using the described framework, shows the results of the topic detection methods and finally also describes the design and outcomes of the automatic speech recognition experiments with domain-specific language models estimated from the collected data.

Keywords Text data mining · Language modeling · Topic identification · Duplicity detection

1 Introduction

Large corpora of text data are the essential prerequisite for building statistical language models (LM) that constitute one of the core components of many applications related to natural language processing (e.g., automatic speech recognition

J. Švec · J. Lehečka · P. Ircing (✉) · L. Skorkovská · A. Pražák · J. Vavruška · P. Stanislav · J. Hoidekr
Department of Cybernetics, Faculty of Applied Sciences, University of West Bohemia, Univerzitní 8, 306 14 Plzeň, Czech Republic
e-mail: ircing@kky.zcu.cz

(ASR), machine translation, OCR, etc.). A decade ago, those corpora were usually assembled "by hand", meaning that they were obtained by transcribing speech recordings from TV and/or radio stations (Psutka et al. 2001), amassed from the existing electronic documents covering a given domain or the language model was trained using a corpus that was originally built for different purposes (Kučera 2002).

It would seem that the problem of data resources has essentially disappeared with the growth of the Internet content as the quantity of electronic texts available on-line nowadays exceed every conceivable limit. However, there are several issues that have to be addressed in order to obtain suitable data for language modeling.

First, it is clear to everybody who has ever visited more than a few Internet pages that the "linguistic" quality of the text is widely variable and that the text gathered from, for example, music fans discussions would not be suitable for training the language model designed for the automatic speech recognition of political debates. Therefore we have to be very careful when selecting the sources of the data. Since we already had specific target domains in mind—namely, the automatic transcription of the parliament sessions and TV political debates (Trmal et al. 2010)—when designing the presented data mining engine, we have decided to collect the data from news websites that were selected as "trustworthy" sources of periodically updated text data.

Extracting the data from news websites results into another important property of the resulting corpus—topicality. The essential property of a suitable language model training corpus is a good coverage of the speech content that is going to be transcribed (or, in other words, low out-of-vocabulary rate on the processed utterances). Words that are not present in the text corpus are usually missing also in the ASR lexicon and consequently cannot be correctly transcribed. The out-of-vocabulary (OOV) rate is therefore a key factor that influences the recognition accuracy. Fortunately, when the language model and the lexicon is built from a sufficiently large corpus, the set of OOV words encountered in TV news and political debates usually consists mostly of personal and geographical names that are related to current events; those are exactly the words that can be extracted from up-to-date Internet news data. This phenomenon is illustrated by the Fig. 1.

The figure shows the number of occurrences of three selected words in particular months starting January 2010. The "Concordia" is the second (and more distinctive) part of the name of the ship "Costa Concordia" that sunk after an accident in January 2012. We dare to claim (although without closer inspection) that previous occurrences of the word "Concordia" were related mostly to other usages of this word (such as the Roman goddess of harmony). Similarly, the surname "Strauss-Kahn" have began to appear much more frequently after the managing director of the IMF Dominique Strauss-Kahn was arrested and charged with a sexual assault in May 2011. According to the graph, the case had quite a steady coverage until September 2011, which coincides with the case dismissal in late August. And finally, even the relative "perennial star" of national news, the former Czech president Václav Havel, has a discernible peaks of attention[1]—the first one in March

[1] Note that the number of occurrences of the word "Havel" is divided by the factor of ten in order to scale down to other two examples.

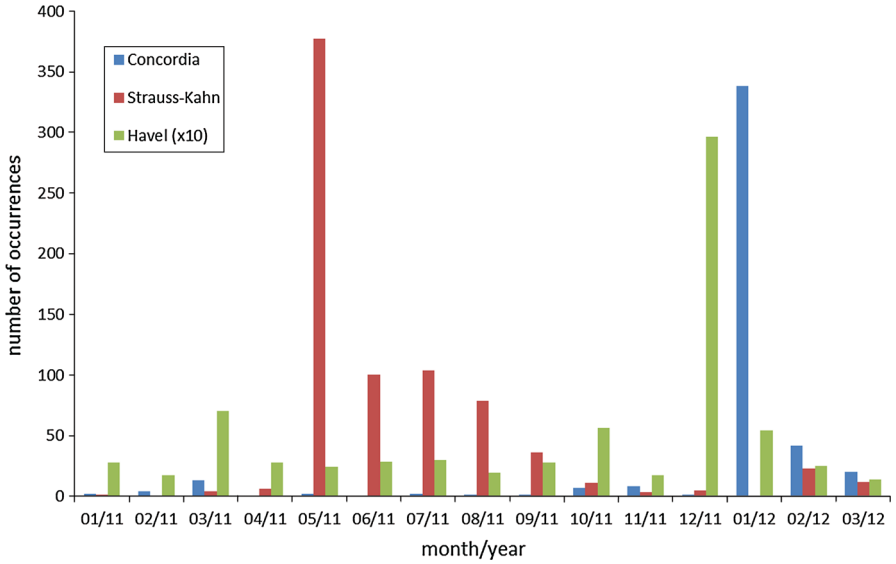

Fig. 1 Occurrences of certain salient words on the timelime

2011 is almost certainly related to the release of his first (and last) movie, the smaller one in October coincides with his birthday and finally the biggest one in December is connected with his death.

Second set of problems related to getting the data from the Internet sources are the more or less technical issues concerning the actual download of the on-line content, the algorithms for stripping of the HTML (or other) markup, methods for text tokenization and normalization and, last but not least, also the detection of possible duplicate documents. The approaches and algorithms dealing with those issues are present in Sects. 3–6.

Once we have the cleaned data from trusted sources available, it is still not practical to use them for language modeling right away. First, the data are typically huge to the extent that it complicates the actual language model construction. Even more importantly, there is the evidence that the data quantity by itself might not be sufficient for good language model performance and what is more important is the right scope of the LM training texts. When the topic of the LM target domain is really specific, it happens that the "in-domain" language model estimated on a moderate-sized corpus vastly outperforms the model built using the data that are one or two orders of magnitude bigger but constitute just a general corpus (Psutka et al. 2003). Thus, when we download and store texts that are meant for future LM training, the information about the document topic is extremely valuable. Such a meta-information is often present in the documents downloaded from the news servers which constitutes another advantage of using this data source. However, the style of marking the topic is usually not consistent across different servers (often not even within a single one) and frequently this information is missing completely. We have therefore decided to use a method for automatic identification of a document

topic that is trained using the topic metadata from the most consistent data source (see Sect. 7). Finally, the statistics of the corpus that we have put together using presented techniques are given in Sect. 8.

2 Related work

There is actually a plethora of articles describing the tools and frameworks for automatic or semiautomatic creation of text corpora from the Web. However, we have found out that most of them aspire to create corpora that are primarily meant for linguistic research in a wide sense and thus they aim for the texts that cover a broad range of topics and consequently yield a representative sample of the general language usage.

With such a goal in mind, it is only natural that the authors usually employ the "wide crawling" approach and use a general search engines (such as Google) or specialized tools like BootCaT (Baroni and Bernardini 2004) to repeatedly submit queries created from the words contained in a "seed list" and collect the retrieved documents. Examples of this approach can be found in Sharoff (2006), Kilgarriff et al. (2010), Li et al. (2007) and, for Czech, Spoustová et al. (2010).

The goal of our work is somehow different. As was already mentioned, our framework is aimed at creating corpora that would adequately and robustly represent only a couple of target domains; consequently, we plan to use them mainly for language model training. Some authors use for the same purposes the "crawling" approach described above (Bulyko et al. 2007), yet we have decided to collect the data from a limited number of "trusted" sources such as news websites. Our system is in this sense similar to the Corporator tool developed by Fairon (2006) but contains more sophisticated processing algorithms, especially for duplicity detection and topic identification.

Interesting work dealing with duplicity detection can be found in Jan Pomikálek's thesis (Pomikálek 2011). He builds upon the work of Broder et al. (1997) just as the algorithms used in our framework. However, he has finished his thesis only recently and we were not aware of his results when we were developing our framework.

3 System architecture

The core of the system is an SQL database on which the individual processing algorithms operate. The design of the architecture used for our data collection and processing engine had to be considered very carefully. Main requirements were the following:

- *Extensibility* The possibility to modify the database scheme according to evolving needs (we expect to use the core engine for the creation of language corpora that might serve for various purposes).
- *Modularity* We always need to be able to add and/or modify algorithms that would perform (sometimes very complex) operations on various databases with

similar inner structure. At the same time, the algorithms should be allowed to invoke each other and they should also be easily configurable.
- *Scalability* The scalability is actually required along two dimensions. First, with regard to the volume of the data—we expect to gather tens of millions of documents. The second dimension involves the possibility of parallelization that would speed up the processing of massive data.
- *Portability* We would like the final system to run on a variety of operational systems and also in several operating modes (interactive, batch, debugging, testing, etc.)

After taking all the requirements into account, we have decided to employ the Voiar library developed by Švec (2010), originally as an efficient platform for spoken term detection in large audiovisual archives (Psutka et al. 2011). The library is implemented in Python and utilizes the SQLAlchemy framework for SQL database access. Extension algorithms can be used for tailoring the Voiar library to a specific task (such as the Web data mining and text processing that we need)—it is possible to add attributes to the objects, define new object types, modify the behavior of existing object, etc. The algorithms also constitute the basic units for parallelization based on MPI (Message Passing Interface).

The overall system architecture is depicted on Fig. 2. The functionality of individual blocks is thoroughly explained in the chapters below.

Once the data are processed with all the algorithms and stored in the database, a wide selection of filters is available in order to export the data into a textual form suitable for language modeling (or other NLP task). Typically, we are able to select the data based on:

- data source (and, in some cases, also a subsource)
- publication date (or period)
- assigned keywords (or the absence thereof)

and more. The individual filters can be combined to create quite sophisticated selections.

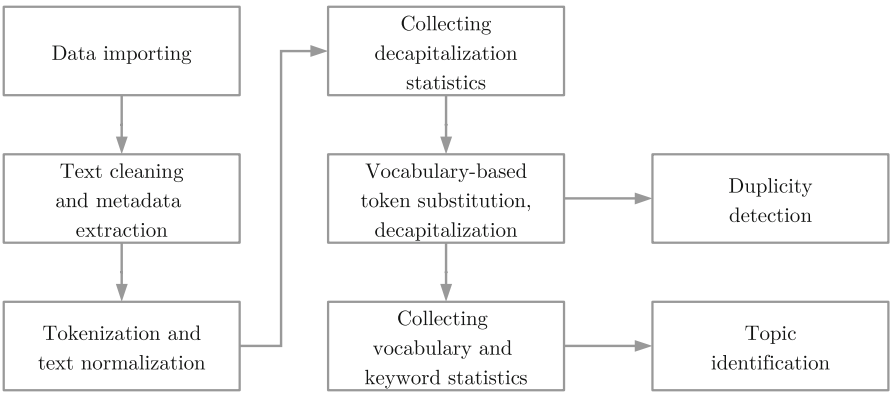

Fig. 2 System architecture schema

4 Data sources

The system database currently contains documents from the following sources:

- *CeskeNoviny.cz* (*CNO*) This website is a news server of the Czech News Agency and as such it provides mostly "serious" home and world news and news from business, culture and sports. It also has the most reliable system for assigning keywords to individual articles and thus the data from this source constitute the training and evaluation corpora for the topic detection algorithms (see Sect. 7).
- *iDnes.cz* (*IDS*) News portal connected with the nationwide newspaper "Mladá fronta DNES". Besides the usual news mix similar to CeskeNoviny.cz it also publishes regional news and hobby-related articles about housing, cars, computers, etc.
- *Lidovky.cz* (*LID*) Internet version of another Czech nationwide newspaper "Lidové noviny". Similar structure as iDnes.cz, probably with less "tabloid-like" articles.
- *ParlamentniListy.cz* (*PAL*) Internet news server dealing predominantly with political issues. Despite its name, it does not have any direct connection with the Czech parliament and is run entirely by a private firm. It publishes a lot of original material and adds diversity to the corpus, while at the same time keeping the focus on politics which is the target domain of many of our ASR applications.
- *Anopress* (*ANP*) These data were actually not acquired online but were provided to us by the Anopress IT a.s., a media monitoring company. The data contain articles published in the printed newspapers (significant portion of those articles is in fact from "Mladá fronta DNES", which causes some overlap with the IDS data) and transcripts of several television news and discussion broadcasts.
- *Otázky Václava Moravce* (*OVM*) The transcripts of the discussion show that were singled out from the ANP data because this broadcast constitutes one of the pilot shows that are currently being live captioned by our ASR system on Czech Television (the national public TV broadcaster) using the shadow-speaker technology (Pražák et al. 2011).
- *Close captions from Czech Television* (*IVY*) This collection contains hidden subtitles that were prepared for the Czech Television broadcasts aired in the period Jan 2000 to Feb 2012.
- *Community-generated subtitles* (*SUB*) Czech subtitles for movies and TV shows created by the online community.

Basic properties of data from all the sources are summarized in Sect. 8, Table 1.

Table 1 Data amounts by data sources

Source	ANP	CNO	IDS	LID	OVM	PAL	IVY	SUB
# articles ($\times 10^3$)	2,226	166	507	78	0.21	52	45	20
# tokens ($\times 10^6$)	580	46	199	31	4	19	84	76
Duplicity ratio (%)	16	14	20	24	0	11	5	38

5 Data preprocessing

Updates of the selected online data sources are periodically checked and downloaded using the standard RSS format. Then the raw (usually HTML-tagged) documents are passed through the cascade of text processing algorithms.

5.1 Text cleaning

The text cleaning algorithm in our system is a rule-based procedure which processes the input web page (an article usually in the HTML format) and extracts the text from the main body of the article. Each of the data sources is assigned a specific set of rules to extract the text and the metadata of the article. The metadata include the date when the article was published, keywords of the article, the author, the title and the subtitle etc. Embedded tables, images and text boxes are excluded from further processing. In addition, the text is checked for invalid characters and character-based substitution is performed.[2] The reduction of the character set simplifies the design of the subsequent processing algorithms.

The source-specific cleaning algorithms become impractical as the number of data sources grows. Thus we have started work on general cleaning algorithm that would be independent of the data source but it was not yet sufficiently evaluated and therefore is not used to process the working data set. The principles of this general algorithm are outlined in Sect. 9.

5.2 Tokenization and text normalization

The task of tokenization algorithm is to segment the input text into the so-called "dictation units", i.e., words, numbers and other characters that are dictated to the ASR system separately. A typical example of such units that have to be found and separated are the punctuation marks which act as standalone dictation tokens (that is, if a user of the ASR system wishes for example the mark "." to be written, he/she has to say "tečka" which means "full stop" in Czech). Unfortunately, especially the full stop character has obviously multiple usages in the written text. It could denote the end of the sentence (in which case the tokenization algorithm separates it from the adjacent word), it could be a part of an abbreviation or it could constitute the decimal mark in a number. In the latter two cases, the full stop remains attached to its neighborhood and the abbreviation and number are replaced with its full-length form and numeral, respectively, that could then be correctly processed by a phonetic transcription module. The correct conversion from numbers to numerals (corresponding word forms) is the main aim of the text normalization module and a non-trivial task for highly inflectional languages such as Czech, since the form of the numeral actually depends on the gender, number and case of the related words (e.g., the number "2" has the correct numeral "dva" if it is in the phrase "2 muži" ("two

[2] For example the Unicode standard defines a special glyph for a ligature "fi". These ligatures are substituted with the sequence of characters "f" and "i".

men") and "dvě" in the phrase "2 ženy" ("two women")). The algorithm based on morphological tagging developed by Zelinka et al. (2005) is used in our system.

5.3 Vocabulary-based token substitution and true casing

Tokens of the normalized text are then processed with a vocabulary-based substitution algorithm—large vocabularies prepared by experts are used to homogenize sequences of tokens. The substitution rules are of three types:

1. Rules for fixing the common typos (they will, for example, replace the misspelled word "zda-li" to "zdali")
2. Rules that replace sequences of tokens with a multiword (e.g., the company name "Czech Coal" is replaced with "Czech_Coal"). Tokens are grouped into multiwords mainly in cases when the meaning of the individual tokens treated separately is quite different from the meaning of the entire multiword. The usage of multiwords makes the language model more accurate and robust, leading to lower perplexity. The rules for creating multiwords correspond predominantly to names of renowned people, political parties and geographical names.
3. The third type of rule unifies the written form of common terms (e.g., a company name "EON" is unified with the correct form "E.ON").

A large number of terms has more than one rule because of inflection. In total, the human-prepared rule lists contain 17k rules.

Another operation called true casing also takes place during the substitution. We use the term true casing to denote the process of substitution of the capitalized words at the beginning of sentences with the corresponding lower-case variants, except for proper names or other word forms commonly written with the capital first letter. The algorithm based on essential corpus statistics was used in the previous version of our data mining framework (Švec et al. 2011). However, it turned out that it incorrectly leaves too many capitalized words unmodified. The most illustrative example is the adjective "český" ("Czech") and its other declensions—according to Czech grammar, it must be written in lowercase except for the case when it constitutes a part of a name (of company, place, etc.). You can image that this adjective is found in many company names in Czech Republic—e.g., "České dráhy" (Czech Railways) or, for that matter, "Česká republika" (Czech Republic) itself. Thus the ratio between the number of occurrences of the capitalized "Český" in the sentence beginnings and in the corpus as a whole (which is used as a decision criterion in Švec et al. 2011) drops bellow a threshold and the word is left unchanged.

Thus we have resorted to a simple rule-base method that essentially takes the capitalized word on possible sentence start (after a punctuation mark) and searches the large vocabulary extracted from Ispell[3] for both capitalized and lower-cased variant. If the lower-cased variant is found while the capitalized is not, the word is decapitalized; otherwise it is left unchanged. That is, it will correctly lowercase the

[3] http://www.cs.hmc.edu/~geoff/ispell.html.

word "Český" mentioned above. Note also that any word that is not present in the Ispell lexicon at all is discarded from our working vocabulary.

6 Duplicity detection

Because of the partial overlap of the data sources (see Sect. 4) and the widespread practice of republishing press material from the news agencies almost unchanged, the database can be expected to contain a substantial number of duplicate documents. There is also a second set of "partial duplicates" resulting from extensive citations from other documents or merging several existing articles into a new one. The detection (and consequent removal) of duplicates is important because the language model created from a text including duplicates can prefer duplicated phrases and sentences instead of correctly modeling the language that is being used in the particular domain.

Our duplicity detection algorithm is based on the shingling method introduced by Broder et al. (1997) and allows us to detect both types of duplicates outlined above. The algorithm first converts each article into a shingle set representation which is composed of a set of overlapping token bigrams.[4] Then the metric rating the similarity of two shingle sets A and B is evaluated. The simplest similarity metric can be defined as a ratio of a number of shingles in both shingle sets to a number of shingles in a union of the two shingle sets:

$$S_1(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

The main disadvantage of this metric (also known as the Jaccard index) is a bad performance in cases where the length of A is very different from the length of B even though $A \subset B$. The solution is to introduce a *containment metric*:

$$S_2(A,B) = \frac{|A \cap B|}{|A|} \tag{2}$$

Unfortunately, the S_2 metric is asymmetric ($S_2(A, B) \neq S_2(B, A)$). Therefore we use the symmetrized *maximum containment metric* defined by Malkin and Venkatesan (2005):

$$S_3(A,B) = \frac{|A \cap B|}{\min\{|A|, |B|\}}. \tag{3}$$

This metric allows us to compare shingle sets with substantially different numbers of elements. The value of S_3 is from the interval [0; 1] where the value 0 means absolutely different shingle sets and the value 1 correspond to the cases where $A \subseteq B$ or $B \subseteq A$.

[4] We have considered using longer token sequences but as processed documents are typically rather short (545 words on average), the usage of higher order n-grams resulted in severe data sparsity.

This definition of duplicity metric allows to define a *duplicity relation*. We say that an article (more precisely a shingle set) A is a *duplicate* of an *original* article B if $S_3(A,B) \geq t_s$ and $S_3(A, B) = S_2(A, B)$ (or, in other words, $|A| < |B|$). Currently we are using $t_s = 0.5$. In other words, the shingle set A is a duplicate of B if there are half or more shingles from A in the shingle set B and the number of shingles in A is lower than the number of shingles in B. For a very rare case $|A| = |B|$ we define that the newer article (according to the date of publication) is the duplicate and the older one is an original.

We can assume that the duplicates occur in a short time window so we detect duplicates only in a set of articles published in a window of two weeks. In the current setup, the detection is performed every day and each run of the detection processes up to 10k articles and takes approximately 7 min. It suggests that the number of evaluations of the Eq. 3 is kept at the acceptable level even though the articles are compared pairwise.

7 Topic identification

As mentioned before, the main purpose of our topic identification module is to filter the huge amount of data according to their topics for the future use as the LM training data. We decided that more than one topic (keyword) should be assigned to each article in our database and that the topics should form some sort of hierarchical system—a topic tree.

7.1 Topic tree

When we started to design our topic identification module, we searched for some kind of an existing topic hierarchy, but we did not find any hierarchical system suitable for our needs. Consequently, we have build our own topic hierarchy in the form of a topic tree, based on our expert findings in topic and keyword distribution in the articles found predominantly in the CNO data and to a smaller extent in the IDS source.

At present the topic tree has 32 main topic categories like health, culture or sports, each of this main category has its subcategories with the "smallest" topics represented as leaves of this tree. An example of a branch representing the topic category justice and courts from the topic tree can be seen on Fig. 3.

In the current system, we use the topic tree with about 450 topics and topic categories, which correspond to the keywords assigned to the articles on the mentioned news servers. The articles with these "originally" assigned topics are used as training data for our identification algorithms.

7.2 Topic identification algorithms

Two methods for automatic topic identification were implemented so far, a classification based on the TF-IDF vector space model and a language-modeling-

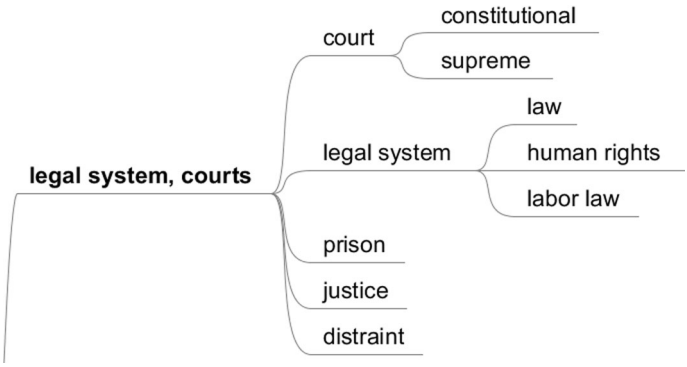

Fig. 3 Branch of the topic tree representing the topic justice and courts (translated from the original Czech version)

based classification. These methods were selected based on their good performance in our previous information retrieval experiments (Kanis and Skorkovská 2010), since we had no experience with the topic identification task so far.

7.2.1 Classification based on language modeling (LM)

The language modeling based approach chosen for the first experiments is similar to the Naive Bayes classifier (Manning et al. 2008), where the probability $P(T|A)$ of an article A belonging to a class (topic in our case) T is computed as

$$P(T|A) \propto P(T) \prod_{t \in A} P(t|T) \tag{4}$$

where $P(T)$ is the prior probability of a topic T and $P(t|T)$ is a conditional probability of a term t given the topic T. This probability can be estimated by the maximum likelihood estimate simply as the relative frequency of the term t in the training articles belonging to the topic T:

$$\hat{P}(t|T) = \frac{tf_{t,T}}{N_T} \tag{5}$$

where $tf_{t,T}$ is the frequency of the term t in T and N_T is the total number of tokens in articles of the topic T.

The goal of this language modeling based approach is to find the most likely topic(s) of an article A, i.e. the ones with the maximum a posteriori probability:

$$T_{map} = \arg\max_T \hat{P}(T|A) = \arg\max_T \hat{P}(T) \prod_{t \in A} \hat{P}(t|T). \tag{6}$$

The prior probability of the topic $\hat{P}(T)$ was implemented as the relative frequency of the articles belonging to the topic in the training set, but we found out that using the uniform prior $\hat{P}(T)$ provides comparable identification results.

7.2.2 Vector space model (VSM) classification

The second tested algorithm is the TF-IDF vector space model based classification. For each term t in the topic T the term frequency $\mathrm{tf}_{t,T}$ and inverse document frequency is computed:

$$idf_t = \log \frac{N}{N_t} \tag{7}$$

where N is the total number of topics and N_t is the number of topics containing the term t. The similarity of an article A and a topic T is then computed as:

$$sim(A, T) = \sum_{t \in A} \mathrm{tf}_{t,T} \cdot idf_t. \tag{8}$$

The topics with the highest similarity are then assigned to the tested article.

8 Corpus analysis and evaluation

8.1 Corpus statistics

This chapter offers some basic statistics and consistency analysis of the gathered corpus, computed after the elimination of duplicate articles.

The number of articles and the number of tokens (without punctuation marks) in particular years of publication is shown in Fig. 4. The bars denoted with *N/A*

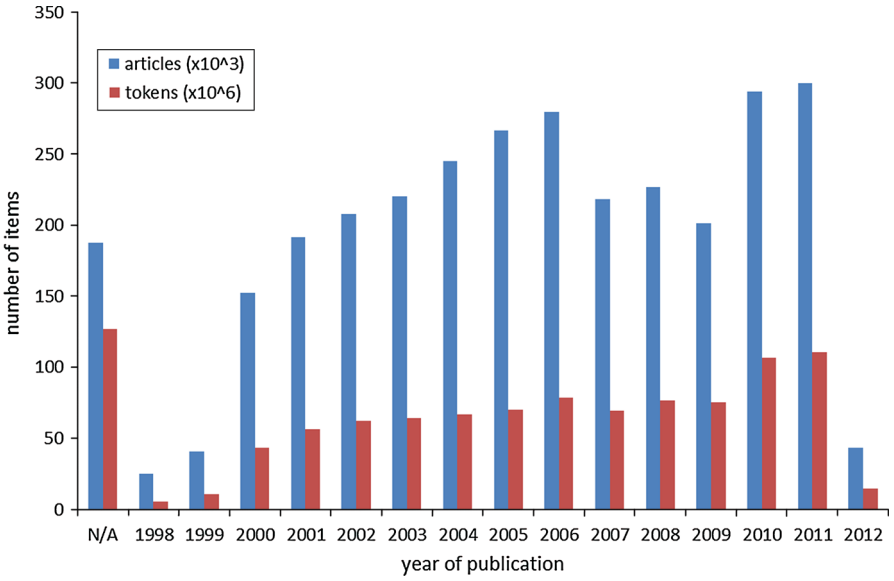

Fig. 4 Data amounts by year of publication

represent all articles for which the publication date was not available. Our corpus currently has more than 3 million articles containing over 1 billion tokens in total.

The number of articles and the number of tokens divided according to the data source is shown in Table 1, along with the duplicity ratio for individual data sources (that is, the percentage of articles that were detected as duplicates and had to be removed from the "raw" data set). It is evident that the ANP source has by far the largest volume of data, which is due to the fact that it contains many "subsources". On the other hand, the OVM source contains only 4 millions tokens, but since this data consist of transcripts of the live discussion show (while the rest of the data are predominantly newspaper articles), it is very valuable for modeling spontaneous speech.

For the comparison of different data sources and for the evaluation of consistency of a particular data source we used a standard Spearman rank correlation coefficient of the distance of ranks of 500 most frequent words (Spoustová et al. 2010; Kilgarriff 2001). It can be seen in Table 2 that all data sources are very consistent (the value on the diagonal is always higher then 0.9). Another interesting observation is the high similarity between ANP, IDS and LID sources. It is most probably caused by the similar newspaper style used in all those data.

The spontaneous speech contained in the OVM data is much different from each of the other sources (the coefficient is always lower than 0.67) which indirectly confirms their importance for building language models usable for automatic transcription of spontaneous speech. That is, the language style of spontaneous speech indeed is substantially different from the style used in newspaper articles. Such finding is further corroborated by the similarity coefficients of IVY and SUB data sources. They are "most similar" to each other (because they are both transcription of spoken content), yet the IVY shows much higher similarity to other sources than SUB. The reason is that a substantial part of IVY data are the transcripts of TV news which cover essentially the same topics as newspaper articles contained in other data sources, whereas the SUB data consist almost entirely of movie a TV shows subtitles.

Figure 5 depicts the dependence of the size of the vocabulary on the number of tokens collected in the corpus. The figure also shows the dependence of the size of a pruned vocabulary which contains only words occurring more than 5, 10, 100 and 1

Table 2 Consistency and similarity coefficients

	ANP	CNO	IDS	LID	OVM	PAL	IVY	SUB
ANP	0.937	0.907	0.976	0.975	0.670	0.858	0.868	0.667
CNO		0.930	0.915	0.946	0.511	0.897	0.653	0.421
IDS			0.954	0.974	0.600	0.826	0.842	0.658
LID				0.950	0.632	0.903	0.800	0.594
OVM					0.911	0.633	0.726	0.626
PAL						0.950	0.629	0.409
IVY							0.987	0.880
SUB								0.994

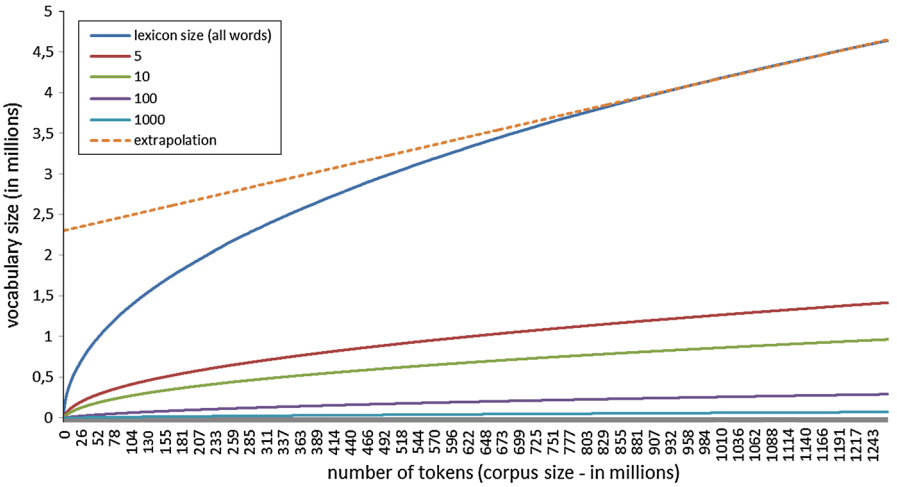

Fig. 5 Dependency of the vocabulary size (pruned to different minimum number of occurrences) on the size of the corpus

000 times, respectively, on the number of tokens. It is evident that the curve representing vocabulary size does not seem to approach a saturation level, i.e., new words keep occurring regardless of the size of the collected data. The straight dashed line represents a linear regression of the curve between 900M and 1,26B tokens and the extrapolation shows that by adding 1 million tokens of text to the corpus the vocabulary grows by about 1,850 words. This fact is in line with the phenomenon illustrated by Fig. 1—especially new proper nouns make its way into the news (and out again) with almost every event, affair or disaster.

8.2 Evaluation of topic detection algorithms

For the evaluation of the individual topic identification methods, a smaller collection of articles from the CNO source was put aside. The articles from *ČeskéNoviny.cz* include keywords that were assigned by their authors (3.5 keywords per article in average). Since the keyword assignment seemed to be rather consistent across authors and articles, we have decided to declare this metadata as the "gold-standard" training and reference topics. This collection contains 158,000 articles, 140,000 of these articles were used as topic training data, remaining 18,000 are available for evaluation testing.

Two types of evaluation were performed on the test collection. The first one is more from the point of view of information retrieval (IR), where each newly downloaded article is considered as a query in IR and precision (P), recall (R) and F_1-measure is computed for the answer topic set:

$$P = \frac{T_C}{T_A}, \quad R = \frac{T_C}{T_R}, \quad F_1 = 2\frac{P \cdot R}{P + R} \tag{9}$$

where T_A is the number of topics assigned to the article, T_C is the number of correctly assigned topics and T_R is the number of relevant reference topics. An average of these measures is then computed across a set of testing articles.

The second type of evaluation is from the point of view of a topic classifier, where P, R and F_1 is computed for each topic separately. Two ways of computing the average measures can be applied in this case, *microaveraging* (topics count proportionally to the size of the topic article set):

$$P_{micro} = \frac{\sum_T T_C}{\sum_T T_A}, \quad R_{micro} = \frac{\sum_T T_C}{\sum_T T_R} \tag{10}$$

and *macroaveraging* (all topics count the same):

$$P_{macro} = \frac{\sum_T P_T}{|T|}, \quad R_{macro} = \frac{\sum_T R_T}{|T|} \tag{11}$$

In this case T_A refers to the number of articles assigned to a topic, T_C is the number of articles correctly assigned to the topic i.e. the "true positives", T_R is the true number of articles with the topic and $|T|$ is the total number of topics. The *macroaverage* measures are more important in our case, because we want our classifier to perform well on infrequent topics, too.

First, we wanted to find out the best number of topics to assign to each article. The relation between the number of topics and P, R and F_1 measures from the IR point of view is shown on Fig. 6; it can be seen that best results are obtained for 3 assigned topics. P, R and F_1 measures obtained for the test set of 18,000 articles and 3 assigned topics are shown in Table 3.

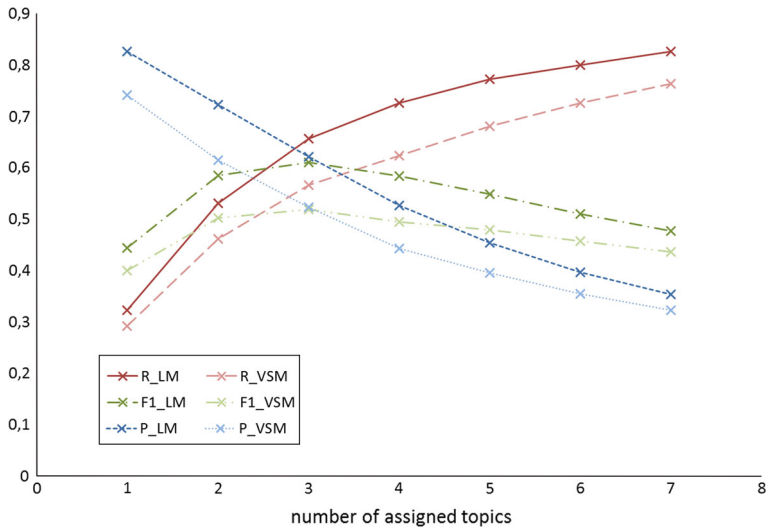

Fig. 6 Dependency of P, R and F1 on the number of assigned topics

Table 3 Average P, R and F_1 measures of topic identification results for the set of 18,000 articles

Method	IR point of view			Microaveraging			Macroaveraging		
	P	R	F_1	P	R	F_1	P	R	F_1
LM	0.594	0.626	0.583	0.597	0.570	0.583	0.624	0.442	0.517
VSM	0.495	0.523	0.486	0.496	0.475	0.485	0.496	0.273	0.352

The language modeling approach seems to achieve better results than vector space modeling, especially for topics with a small article set, which can be seen from the *macroaverage* R and F_1 measures.

It may seem at the first glance that the results are not particularly good, but it must be taken into consideration that we have a very large set of topics that are in many cases not well distinguished. Also the articles in the test collection are taken as they were on the news server, the original reference topics was not revised in any way, so in many cases the topic we assign to the article is also "correct", but it is not included in the reference set of topics. For example, the article about the achievements of the hockey representation has only hockey in reference topics, but our topic identification module assigned the topics hockey, representation, which is correct as well.

Finally, the Fig. 7 shows the number of articles that fall within particular first-tier branches of our topic tree (that is, articles that are tagged with any of the keywords belonging to a subtree with the given headword).

8.3 Language modeling and ASR experiments

The main motivation for the development of an automatic topic identification method introduced in Sect. 7 was that we wanted to be able to effectively retrieve large amounts of domain-specific data for language model training. In this chapter, we will therefore present several experiments with language models estimated on the text corpora that were filtered from the large database of newspaper articles using various selection criteria. Since the ultimate measure of the language model quality is the performance of the system where the LM is employed (in this case the ASR decoder), we will also describe the speech recognition system that we have used and report relevant Word-Error-Rates (WER).

All language models perplexities (PPL) and WER in the first set of experiments were evaluated on a test set consisting of speech obtained during the testing phase of the automatic closed-captioning system that employs the so-called "shadow-speaker" approach (Pražák et al. 2011). It means that the potentially noisy and/or overlapping broadcast speech is respoken with a trained speaker in controlled acoustic conditions in order to ensure higher recognition accuracy. The evaluation set contains recordings from just a single female speaker. The total length of the first test set audio is 98 min.

Since the speaker, whose utterances are in the test set, recorded in fact over 25 h of data in total, we were able to tailor the acoustic models of the ASR system to this particular speaker (see Vaněk and Psutka 2010 and Zajíc et al. 2010 for details).

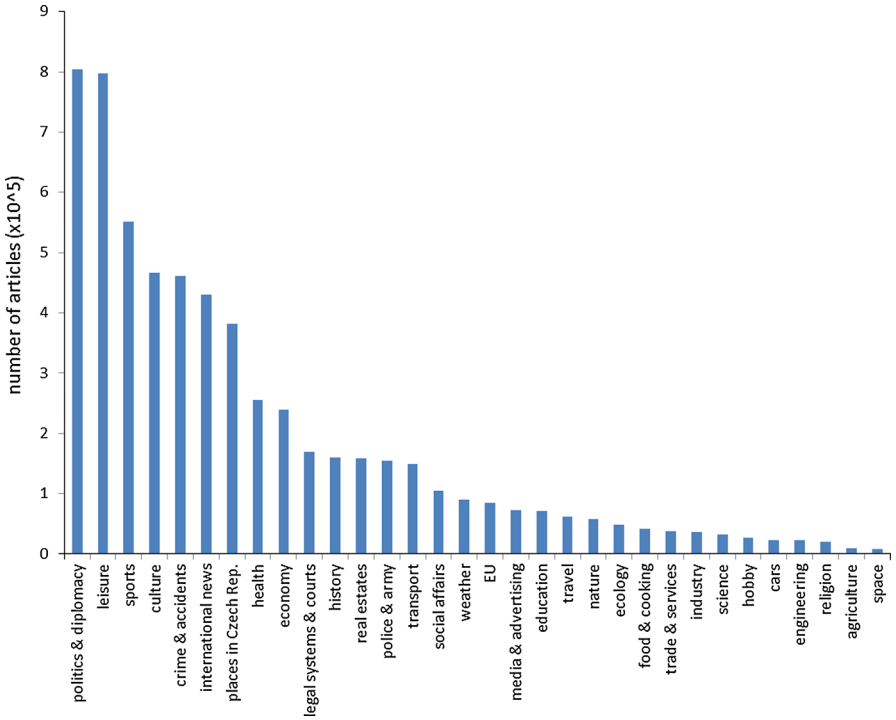

Fig. 7 Histogram of the first-tier topics

This gives us a very high quality acoustic model and consequently, we can safely assume that any ASR performance gain from the improved language model would be even more prominent in the case when the acoustic model is less effective. All the language models described in the following paragraphs are trigram LMs estimated using the SRI Language Modeling Toolkit (SRILM) by Stolcke (2002) employing the default Good-Turing discounting method. The resulting models always contain all the lexicon word bigrams that are found in the training data; the trigrams must occur at least twice to be included in the model.

The first test set consists of samples of the dialogues that took place during the political talk show ("Otázky Václava Moravce"—cf. the OVM data) broadcast by the Czech Television on July 18th, 2010. The first line of Table 4 thus shows results for the baseline model estimated from all articles published between year 1998 (date of the oldest articles contained in our database) and July 17th, 2010. Note that the model is very large, with a vocabulary exceeding one million words.

The show whose part constitutes our test set discussed mainly the newly appointed Czech government, the state budget and also health care issues. The appropriate keywords from the first tier of the tree would then be politics and diplomacy (which we will denote as politics from now on), economy and health.[5] The

[5] Note that assuming to know the topics before the actual broadcasting is not unrealistic—the main "themes" of each debate are published on the broadcaster website beforehand.

Table 4 Properties of language models trained using different data selections—OVM test set

Selection ID	# tokens	Vocab size	LM size (MB)	OOV (%)	PPL	WER (%)
All data till 07/17/2010	690M	1,018k	4,571	0.56	629	5.03
Pol.	205M	628k	1,575	0.62	571	5.29
Pol.+econ.	247M	661k	1,826	0.62	551	5.13
Pol.+econ.+heal.	316M	780k	2,327	0.60	559	5.13
Sport	132M	436k	865	1.57	3,848	8.65

following three lines of the table thus present the results for the language models that were estimated from the filtered set of articles from the baseline corpus—only the articles labeled with any keyword that comes from the subtree with the headword politics, politics and economy, and politics, economy and health, respectively, were included in the selection.

With the topic-filtered models, we observed moderate drop in the recognition performance (2–5 % relative) but significant reduction of the language model size (49–65 %) which is a factor that might not be that important in the laboratory setting but plays a crucial role when designing a system that is supposed to work for example on a portable device with limited hardware resources.

Analyzing the results on the OVM test set, we have also came up with a hypothesis that such a discussion show for general audience does not really contain much of a domain-specific sublanguage to benefit from the topic-specific models. In order to test this hypothesis, we have constructed a second test set containing three broadcasts of commented tennis matches from the 2012 Summer Olympics (total length 2:15 h). The results are summarized in Table 5. This time the general language model is even bigger that in the case of the OVM data, because the first of the matches took place on July 28, 2012, and we have included all the articles up to the day before this match. It can be seen that the ASR performance is in general much worse—it is due to the fact that we have used the acoustic models trained for a completely different acoustic channel and also because of the noisy background present in the audio (this time there was no re-speaking—the speech was taken directly from the audio broadcasted from the event venue). However, the performance gain was observed when restricting the data first to the sports articles (first-tier topics) and then specifically to articles labeled with tennis. Note that the tennis language model is almost a hundred times smaller than the general one.

Since the adverse acoustic conditions and mismatching acoustic channel in the tennis test set rendered the results unconvincing, we have prepared yet another test

Table 5 Properties of language models trained using different data selections—tennis test set

Selection ID	# tokens	Vocab size	LM size (MB)	OOV (%)	PPL	WER (%)
All data till 07/27/2012	862M	1,073k	5,499	15.45	829	68.86
Sports	160M	464k	1,059	15.20	690	67.72
Tennis	6M	96k	56	13.47	611	67.44

Table 6 Properties of language models trained using different data selections—"Sandy" test set

Selection ID	# tokens	Vocab size	LM size (MB)	OOV (%)	PPL	WER (%)
All data till 11/02/2012	870M	1,075k	5,283	2.97	594	13.63
Crime and accidents	105M	519k	858	2.97	641	16.25
Weather	20M	246k	194	3.36	631	16.73
International news	145M	648k	1,206	2.97	795	15.73
Catastrophes	2.3M	81k	30	4.52	453	18.67
Disasters	14M	209k	142	4.39	1,281	22.42
Storms	0.4M	27k	6	8.53	406	24.97
Americas, USA	88M	567k	814	2.97	891	16.51
Interpolation	870M	1,075k	5,286	2.97	282	12.73

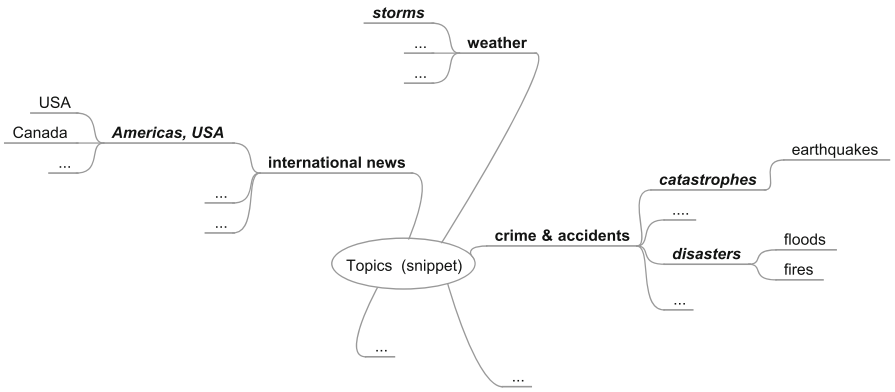

Fig. 8 Section of the topic tree showing the first-tier and second-tier topics used for the "Sandy" test set

set. It consists of a news article reporting the situation after the strike of the hurricane Sandy. This article, containing 774 words, was read by 4 speakers and recorded using a high-quality headset. The language models were again prepared first using the all the collected data up to the date of the article publication (line 1 of Table 6). Then we have estimated 3 topic-specific models using the broader first-tier topics (lines 2–4) and another 4 using more specific second-tier topics (lines 5–8). The relation between individual topics within the topic tree is shown in Fig. 8.

The results presented in Table 6 show that none of the topic-specific language models alone had outperformed the model built from all the data. We suspect that the reason could be that the selected article covers in fact several topics spread across the topic tree. We have therefore decided to interpolate all the models listed in the table (including the one from the entire dataset). The interpolation coefficients were determined in an unsupervised manner—first, the test set was recognized using the general model from all the data, then the recognition output was used as a "development set" for tunning the interpolation parameters using the compute-best-mix tool from SRILM Stolcke (2002) and finally the interpolated model was

employed to re-recognize the test data. That way we have managed to reduce the WER by 6.6 % relative.

9 Discussion and work in progress

We have presented a flexible and scalable framework for downloading, processing and storing large amounts of electronic text data that could be then used for various purposes related to natural language and speech processing. Main goal of our work was to develop a system that would allow to automatically create different subcorpora mainly using time- and topic-specific filters—as our results show, this goal was successfully met.

The framework is currently routinely used within our research team for building suitable text corpora that are used in speech recognition systems prepared for several domains. The layout and algorithms used in the framework are generally language independent; however, there are several modules that make use of the Czech-specific resources, such as the vocabularies for token substitution and true casing and also the topic tree. Those need to be replaced when porting the system to a different language environment.

However, there are still some functionalities of the system that we have found worth implementing and are currently a subject to evaluation experiments.

The first of them is a general algorithm for cleaning the content downloaded from an arbitrary Web data source. The current version of the tool employs a set of multiple rule-based cleaning algorithms, each of them tailored to a specific data source (see Sect. 5.1). This approach becomes impractical with the growing number of sources. The devised general algorithm first removes from the downloaded item all the chunks that could be detected as not being a part of the article text on the basis of the HTML tags (e.g., hyperlinks, lists, tables, pictures, etc.). Then the resulting text is split into paragraphs that are consequently classified as belonging or not belonging to the article text using a k-means classifier. The features of the classifier include for example the number of OOV words in the given paragraph, number of hyperlinks, punctuation marks, etc.

The second investigated issue is related to detecting salient new words from the downloaded content. It is impossible to add all the words from the corpus to the lexicon of the ASR decoder due to the limitation stemming from the computational demands of the decoding process,[6] yet it is highly desirable to add those new words that are likely to be used frequently in the upcoming period (let us refer once again to Fig. 1.). We have designed a semi-automatic procedure for selecting candidate words for the vocabulary extension. First, all the new words are filtered based on their capitalization (as the capitalized words are more likely to be salient) and their frequency distribution across the timeline. The preselected list is then presented to

[6] Please note that even though our decoder can handle a lexicon with up to one million words (which makes it one of the world's best in this aspect), it is still not able to accommodate all the words occurring in our corpora, not even just the ones that occurred at least five times—see Fig. 5.

the human annotator that makes the final decision about adding words to the ASR lexicon.

Finally, the framework is being adapted for handling audiovisual data, with a special focus on the data that are equipped with text subtitles capturing the content of the audio track. This extension will allow us to store also the material that could be later used for training acoustic models or potentially also processed with algorithms dealing with the visual component.

Acknowledgements This work has been supported by the grant of The University of West Bohemia, project No. SGS-2010-054 and by the Grant Agency of the Czech Republic, project No. GAČR P103/12/G084. The access to the MetaCentrum computing facilities provided under the programme Projects of Large Infrastructure for Research, Development, and Innovations LM2010005 funded by the Ministry of Education, Youth, and Sports of the Czech Republic is appreciated.

References

Baroni, M. & Bernardini, S. (2004). Bootcat: Bootstrapping corpora and terms from the web. In *In Proceedings of LREC 2004*, pp. 1313–1316.

Broder, A. Z., Glassman, S. C., Manasse, M. S., & Zweig, G. (1997). Syntactic clustering of the web. *Computer Networks and ISDN Systems, 29*(8–13), 1157–1166.

Bulyko, I., Ostendorf, M., Siu, M., Ng, T., Stolcke, A., & Çetin, O. (2007). Web resources for language modeling in conversational speech recognition. *ACM Transactions on Speech and Language Processing (TSLP), 5*(1), 1:1–1:25.

Fairon, C. (2006). Corporator: a tool for creating rss-based specialized corpora. In *Proceedings of the 2nd international workshop on web as corpus, WAC '06* (pp. 43–49). Stroudsburg, PA, USA: Association for Computational Linguistics.

Kanis, J., & Skorkovská, L. (2010). Comparison of different lemmatization approaches through the means of information retrieval performance. In: P. Sojka, A. Horák, I. Kopeček, & K. Pala (Eds.), *TSD 2010. LNCS* (Vol. 6231, pp. 93–100). Heidelberg: Springer.

Kilgarriff, A. (2001). Comparing corpora. *International Journal of Corpus Linguistics, 6*(1), 97–133.

Kilgarriff, A., Reddy, S., Pomikálek, J., & PVS, A. (2010). A corpus factory for many languages. In N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, & D. Tapias (Eds.), *Proceedings of the seventh international conference on language resources and evaluation (LREC'10)* (pp. 904–910). Valletta, Malta: European Language Resources Association (ELRA).

Kučera, K. (2002). The Czech National Corpus: Principles, design, and results. *Literary and Linguistic Computing, 17*(2), 245–257.

Li, P., Zhu, Q., Qian, P., & Fox, G. (2007). Constructing a large scale text corpus based on the grid and trustworthiness. In: V. Matousek & P. Mautner (Eds.), *TSD. Lecture Notes in Computer Science* (Vol. 4629, pp. 56–65). New York: Springer.

Malkin, M. & Venkatesan, R. (2005). Comparison of texts streams in the presence of mild adversaries. In *Proceedings of the 2005 Australasian workshop on grid computing and e-research* (Vol. 44, pp. 179–186). ACSW Frontiers '05. Australian Computer Society, Inc.,.

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. New York: Cambridge University Press.

Pomikálek, J. (2011). *Removing boilerplate and duplicate content from web corpora*. Ph.D. thesis, Faculty of Informatics, Masaryk University, Brno, Czech Republic.

Pražák, A., Loose, Z., Psutka, J., Radová, V., & Müller, L. (2011). Four-phase re-speaker training system. In *Proceedings of SIGMAP 2011*. Seville.

Psutka, J., Ircing, P., Psutka, J.V., Radová, V., Byrne, W., Hajič, J., Mírovský, J., & Gustman, S. (2003). Large vocabulary ASR for spontaneous Czech in the MALACH project. In *Proceedings of Eurospeech 2003* (pp. 1821–1824). Geneva.

Psutka, J., Radová, V., Müller, L., Matoušek, J., Ircing, P., & Graff, D. (2001). Large broadcast news and read speech corpora of spoken Czech. In *Proceedings of Eurospeech 2001* (pp. 2067–2070). Denmark: Aalborg.

Psutka, J., Švec, J., Psutka, J.V., Vaněk, J., Pražák, A., Šmídl, L., & Ircing, P. (2011). System for fast lexical and phonetic spoken term detection in a Czech cultural heritage archive. *EURASIP Journal on Audio, Speech, and Music Processing*, 10.

Sharoff, S. (2006). Creating general-purpose corpora using automated search engine queries. In *WaCky! Working papers on the Web as Corpus* (pp. 63–98). Gedit.

Spoustová, D., Spousta, M., & Pecina, P. (2010). Building a Web Corpus of Czech. In *Proceedings of the seventh conference on international language resources and evaluation (LREC'10)*. Valletta, Malta.

Stolcke, A. (2002). SRILM—an extensible language modeling toolkit. In *Proceedings of ICSLP 2002* (pp. 901–904). Denver.

Švec, J. (2010). The Voiar (Voice Archive) library. University of West Bohemia, Plzeň.

Švec, J., Hoidekr, J., Soutner, D., & Vavruška, J. (2011). Web text data mining for building large scale language modelling corpus. In: I. Habernal & V. Matoušek (Eds.), *Text, speech and dialogue*. *Lecture Notes in Computer Science* (Vol. 6836, pp. 356–363). Berlin / Heidelberg: Springer.

Trmal, J., Pražák, A., Loose, Z., & Psutka, J. (2010). Online TV Captioning of Czech Parliamentary Sessions. In: Sojka, P., Horák, A., Kopeček, I., & Pala, K. (Eds.), *Text, speech and dialogue*. *Lecture Notes in Artificial Intelligence* (Vol. 6231, pp. 416–422). Berlin: Springer.

Vaněk, J. & Psutka, J. (2010). Gender-dependent acoustic models fusion developed for automatic subtitling of parliament meetings broadcasted by the Czech TV. In: P. Sojka, A. Horák, I. Kopeček, & K. Pala (Eds.), *TSD 2010*. *LNCS* (Vol. 6231, pp. 431–438). Heidelberg: Springer.

Zajíc, Z., Machlica, L., & Müller, L. (2010). Robust statistic estimates for adaptation in the task of speech recognition. In: P. Sojka, A. Horák, I. Kopeček, & K. Pala (Eds.), *TSD 2010*. *LNCS* (Vol. 6231, pp. 464–471). Heidelberg: Springer.

Zelinka, J., Kanis, J., & Müller, L. (2005). Automatic transcription of numerals in inflectional languages. In: V. Matoušek, P. Mautner, & T. Pavelka (Eds.), *Text, speech and dialogue*. *Lecture Notes in Computer Science* (Vol. 3658, pp. 326–333). Berlin/Heidelberg: Springer.