Slovak Web Discussion Corpus

Daniel Hládek, Ján Staš, Jozef Juhár

daniel.hladek@tuke.sk, jan.stas@tuke.sk, jozef.juhar@tuke.sk
Department of Electronics and Multimedia Communications
Technical University of Košice
Slovakia

Abstract

The corpus makes it possible to study spontaneous, interactive communication that often includes various incorrect or unusual words. The corpus includes an index for easy searching using regular expressions. Text of the discussions is processed using our tools for word tokenization, sentence boundary detection and morphological analysis. Token annotations include a correct word, proposed by a statistical correction system.

Corpus Contents

This contribution aims to provide a representative sample of Slovak colloquial language in an organized corpus. The corpus includes a complete set of web discussions about various topics from a single site. Each discussion is marked with a topic and talking person and is assigned to a section.

Contents are sorted according to discussion topic.

Section	# tokens	sentences	items
philosophy	94 057	11 127	41
culture	42 214	4 700	29
relationships	1 504 175	175 796	881
religion	2 546 546	239 079	834
computers	139 275	16 488	210
politics	808 724	79 195	552
miscellaneous	4 646 054	511 600	3320
sport	11 614	1 257	10
health	38 727	4 463	17
Together	**9 831 274**	**1 043 705**	**5 894**

Word and Sentence Boundary Detection

The main goal is to distinguish between types of tokens that are interesting for further processing by adding and removing spaces and unnecessary characters as it is required.
The following types of tokens are recognized:

- words and acronyms,
- abbreviations,
- various number representations,
- URLs and e-mails,
- punctuation.

1. List of recognized tokens is searched. The longest matching token is selected.
2. If recognized token is a dot, colon, empty line, exclamation mark or question mark, the end of sentence is found.
3. If no token is found, the first character is discarded an the search process continues.
4. If some other token is found, it is added to the sentence, characters are discarded from the input and the search process continues. If the token is the first in the sentence and it is not in the list of exceptions then it is lowercased.
5. If there are no more characters in the input string, the search process finishes.

Freely available at http://nlp.web.tuke.sk

the Corpus, papers, processing tools, web search, web language processing demo, other resources

Index File with Annotations

```
corpus/náboženstvo/náboženstvo00006.txt  Teológia - ako filozofia ?   3. mája 2010, 09:03          Ónya
corpus/politika/politika00007.txt        Sklamania ... a zase         19. júla 2010, 20:50         dnes_flamujem
corpus/rôzne/rôzne00008.txt              Pre istotu                   17. novembra 2013, 19:36 Shadow925
corpus/náboženstvo/náboženstvo00010.txt  Prečo som ateista?            9. januára 2012, 00:02      Lemmya
```

Corpus Gathering

A specialized web agent is used to explore the discussion web site. After HTML code is downloaded, content of each web page is analyzed and saved in a database. A custom parser is designed to extract interesting meta-information (described below) about each discussion. Discussions are sorted into sections according to their theme, as it was found on the web site

http://diskusneforum.sk/

Annotated Document

```
anikaaa|%|SSfs1|% 90|<INT>|%|% vraví|%|VKesc+|% :|%|Z|%
poradíte|%|VKdpb+|% mi|<STOP>|PPhs3|mi ako|<STOP>|O|% sa|<STOP>|R|% mam|%|SSis1|
mám naucit|%|SSis1|naučiť blogovat|%|SSis1|% ?|%|Z|%

lubos|%|%|% 250|<INT>|%|% vraví|%|VKesc+|% :|%|Z|%
tu|<STOP>|PD|% som|<STOP>|VKesa+|% o|<STOP>|Eu6|o tom|<STOP>|PFns6|% nieco|%|
PFns4|niečo

Vlado|<MP1>|%|% vraví|%|VKesc+|% :|%|Z|%
a|<STOP>|O|% čo|<STOP>|PFns4|% potrebuješ|%|VKesb+|% blokovať|%|VIj+|% ?|%|Z|%
nervy|%|SSip4|% ?|%|Z|%
verejnú|%|AAfs4x|% dopravu|%|SSfs4|% ?|%|Z|%
zápchu|%|SSfs4|% ?|%|Z|%

domin|%|SSis1|% vraví|%|VKesc+|% :|%|Z|%
ja|<STOP>|PPhs1|% znam|%|VKesa+|znám .|%|Z|%

ruwolf|%|SSms1|% vraví|%|VKesc+|% :|%|Z|%
a|<STOP>|O|% čo|<STOP>|PFns1|% na|%|Eu4|na to|<STOP>|PFns4|% nevieš|%|VKesb-|% ?|%|
Z|%
všade|%|PD|% máš|%|VKesb+|% návody|%|SSip1|% -|%|Z|% ak|%|O|% nevieš|%|VKesb-|% po|
<STOP>|Eu6|% EN|%|W|% ,|%|Z|% bloguj|%|VMdsb+|% na|%|Eu6|na slovenskom|%|AAis6x|%
a|<STOP>|O|a bo|%|O|% českom|%|AAis6x|% .|%|Z|%
```

Available Annotations

Token boundary identification
Sentence boundary identification
Morphological Analysis
Named Entity Recognition
Automatic Correction

Corpus Usage

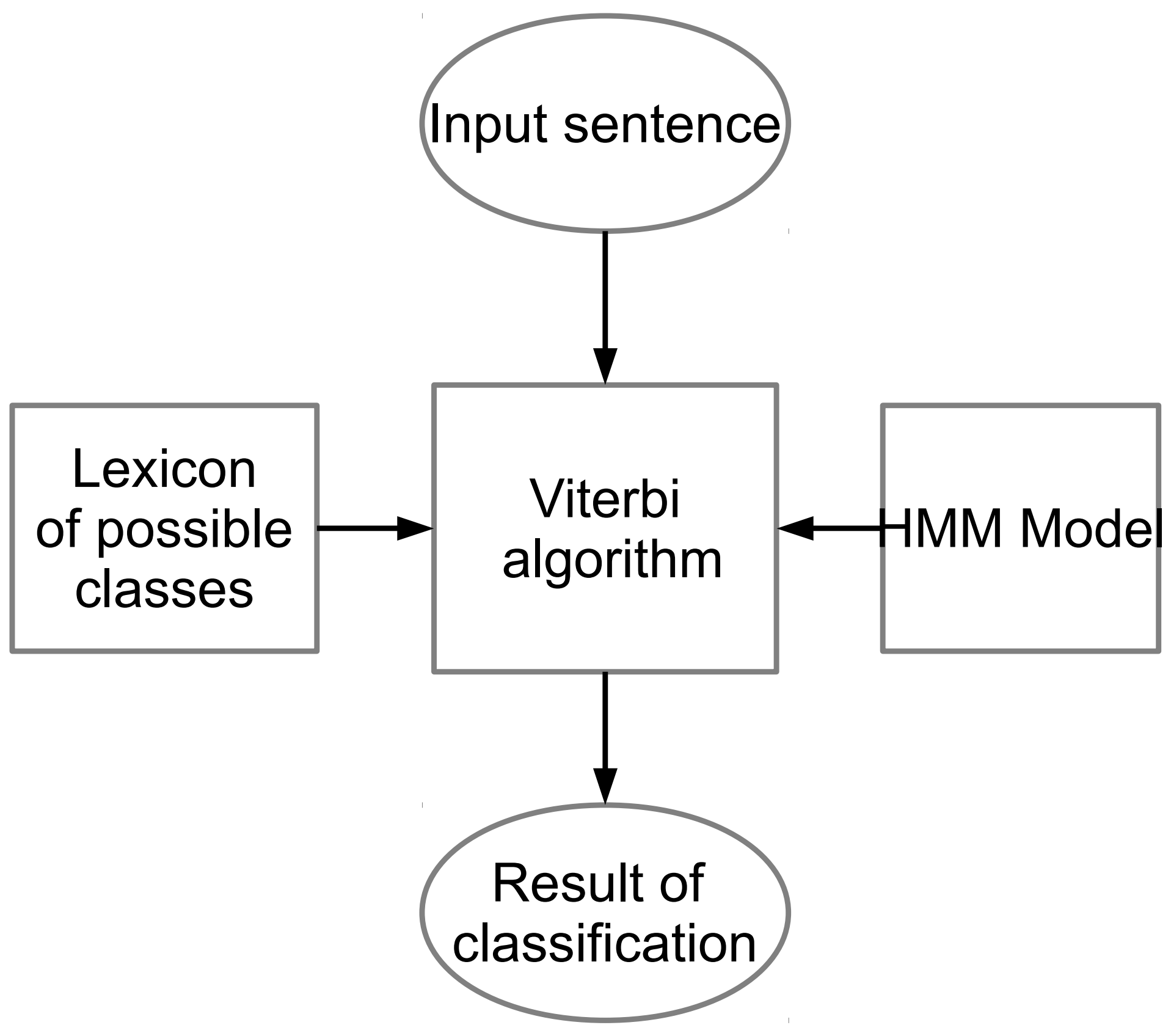
The proposed form and annotations should enable further classical and computational linguistic research of a contemporary way of communication - web discussions. Its size should be sufficient for statistical analysis of word connotations, language modeling or document classification, clustering or information retrieval tasks. Future effort will be focused on processing data from social networks.

Automatic Error Correction

The most common error is incorrect typing of word, where certain letters are replaced with incorrect equivalents with diacritical marking removed. In the case of the Slovak language, it is possible that one incorrect form can have more possible correct forms. The incorrect words are still readable and are recognizable to a human reader, but automated processing requires some kind of disambiguation that can distinguish the correct form of a word meant by the author.
The lexicon of possible corrections is created by taking a large vocabulary of correct words and for each word a set of possible incorrect forms is generated.
The list of possibly incorrect words is generated using a formal grammar. The most common errors in the written Slovak are omission of diacritical markings (such as n, or a) and incorrect usage of letters y an I.
For each word containing one or more letters that can be typed incorrectly, a list of all possible incorrect forms is generated by recursive application of rewrite rules.
The second part of the correction system is a statistical language model trained on a corpus of texts that are considered to be well written and correct.

HMM-Based Classifier

A HMM-based classifier takes the lexicon and the language model into account and Viterbi algorithm is used to find the most probable sequence of correct words for the given sentence that can contain misspellings.



Input sentence

Lexicon of possible classes

Viterbi algorithm

HMM Model

Result of classification

Morphological Analysis

The most important part of the annotation process is the morphological annotator Dagger.
This classifier uses second-order hidden Markov model and Viterbi algorithm and can utilize grammatical features for smoothing of the observation and transition matrix for improvement classification accuracy.
The model has been trained on trigram counts from the Slovak National Corpus and uses their tag set containing 3500 distinct tags. The search space of the classifier is restricted by a lexicon that contains a list of possible tags for each known word. Observation probabilities are smoothed using custom algorithm that takes morphological features of words into account. The classifier is 86% correct.

We support research activities in Slovakia / This project is being co-financed by the European Union

Acknowledgements

The research presented in this paper was supported by Research and Development Operational Program funded by the ERDF under the project numbers ITMS-26220220182 (50%) and ITMS-26220220141 (50%).

PolTAL 2014: The 9th Conference on Natural Language Processing, 17-19.9.2014 Warsaw, Poland