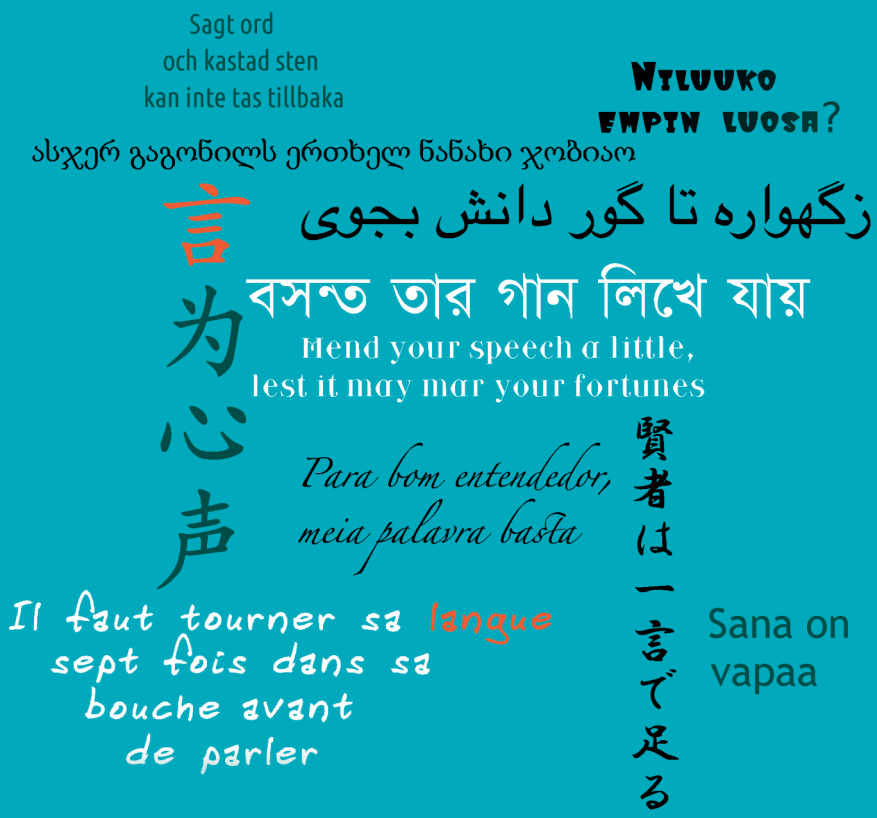
Department of Information and Computer Science

Language- and domain-independent text mining

Mari-Sanna Paukkeri



Aalto University

DOCTORAL
DISSERTATIONS

Aalto University publication series
DOCTORAL DISSERTATIONS 137/2012

Language- and domain-independent text mining

Mari-Sanna Paukkeri

Doctoral dissertation for the degree of Doctor of Science in Technology to be presented with due permission of the Aalto University School of Science, for public examination and debate in Auditorium AS1 of the school on 9th November 2012 at 12 noon.

Aalto University
School of Science
Department of Information and Computer Science

Supervising professor
Prof. Erkki Oja

Thesis advisors
Doc. Timo Honkela
Dr. Mathias Creutz

Preliminary examiners
Dr. Reinhard Rapp, Johannes Gutenberg University Mainz, Germany
Dr. Roman Yangarber, University of Helsinki, Finland

Opponent
Doc. Jussi Karlgren, Gavagai AB, Sweden

Aalto University publication series
DOCTORAL DISSERTATIONS 137/2012

ISBN 978-952-60-4833-8 (printed)
ISBN 978-952-60-4834-5 (pdf)
ISSN-L 1799-4934
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)
http://urn.fi/URN:ISBN:978-952-60-4834-5

Unigrafia Oy
Helsinki 2012

Finland

441 697
Printed matter

Aalto University

Abstract

Aalto University, P.O. Box 11000, FI-00076 Aalto www.aalto.fi

Author
Mari-Sanna Paukkeri

Name of the doctoral dissertation
Language- and domain-independent text mining

Publisher School of Science

Unit Department of Information and Computer Science

Series Aalto University publication series DOCTORAL DISSERTATIONS 137/2012

Field of research Computer and Information Science

Manuscript submitted 4 May 2012 **Date of the defence** 9 November 2012

Permission to publish granted (date) 11 September 2012 **Language** English

☐ **Monograph** ☒ **Article dissertation (summary + original articles)**

Abstract

The field of natural language processing (NLP) has developed enormously during the last decades. The availability of constantly increasing amount of textual data in electronic form has accelerated also the development of statistical methods for NLP, in which characteristics of natural languages are learned from large corpora. Statistical methods have shown their applicability in information retrieval, in which documents of various languages and domains are returned according to user queries, statistical machine translation which is easily applicable to new languages, document clustering to group semantically similar documents, and many information extraction tasks, including keyphrase extraction, document summarization and discovering linguistic features. However, a majority of the NLP research, including also many statistical methods, is concentrated on the English language, using various language-specific tools and resources, such as part-of-speech taggers and ontologies, which are not directly applicable to other languages. Furthermore, methods developed for English alone may not be suitable for languages with different syntax or writing system.

In this dissertation, language-independent methods for natural language processing are developed and discussed. Language-independent methods can be applied to a variety of languages without requiring additional language-specific resources. Also dialects, historical forms of languages, languages of few speakers and languages used in specific domains are accessible with language-independent methods.

As the main contribution of this thesis, *Likey*, a language-independent method for keyphrase extraction and feature selection is developed. The method is applied to keyphrase extraction from encyclopedias and scientific articles in eleven languages, and further used as a feature selection method for automatic taxonomy learning and in a novel approach to user modelling in document difficulty assessment. Another major contribution is related to document representations: a set of dimensionality reduction and distance measures are compared in a document clustering task, a novel language-independent direct evaluation method for document representations is proposed, and linguistic features are used for document clustering in a lexical choice task.

Keywords natural language processing, computational linguistics, unsupervised machine learning, language independence, subjectivity of language use, keyphrase extraction, document clustering

ISBN (printed) 978-952-60-4833-8 **ISBN (pdf)** 978-952-60-4834-5

ISSN-L 1799-4934 **ISSN (printed)** 1799-4934 **ISSN (pdf)** 1799-4942

Location of publisher Espoo **Location of printing** Helsinki **Year** 2012

Pages 250 **urn** http://urn.fi/URN:ISBN:978-952-60-4834-5

Aalto-yliopisto

Tiivistelmä

Aalto-yliopisto, PL 11000, 00076 Aalto www.aalto.fi

Tekijä
Mari-Sanna Paukkeri

Väitöskirjan nimi
Kielestä ja aihealueesta riippumaton tekstinlouhinta

Julkaisija Perustieteiden korkeakoulu

Yksikkö Tietojenkäsittelytieteen laitos

Sarja Aalto University publication series DOCTORAL DISSERTATIONS 137/2012

Tutkimusala Informaatiotekniikka

Käsikirjoituksen pvm 04.05.2012 **Väitöspäivä** 09.11.2012

Julkaisuluvan myöntämispäivä 11.09.2012 **Kieli** Englanti

☐ **Monografia** ☒ **Yhdistelmäväitöskirja (yhteenveto-osa + erillisartikkelit)**

Tiivistelmä

Luonnollisen kielen käsittely (*Natural language processing*, NLP) on tieteenalana kasvanut valtavasti viimeisinä vuosikymmeninä. Tekstimuotoista tietoa on tarjolla eletronisessa muodossa jatkuvasti enenevässä määrin. Tämä on kiihdyttänyt myös tilastollisten NLP-menetelmien kehitystä, joissa kielen ominaisuuksia opitaan automaattisesti suurista tekstiaineistoista. Tilastollisia menetelmiä on onnistuneesti sovellettu tiedonhakuun, jossa käyttäjän hakusanojen perusteella palautetaan dokumentteja eri kielillä ja eri aloilta, tilastolliseen konekäännökseen, jota pystytään helposti laajentamaan uusiin kielipareihin, dokumenttien klusterointiin, jossa merkityssisällöltään samankaltaiset dokumentit ryhmitellään yhteen, ja moniin tiedonirrotustehtäviin, kuten avainfraasien hakuun, tekstin referointiin ja tiivistämiseen sekä kielitieteellisten piirteiden hakuun. Valitettavasti suurin osa NLP-tutkimuksesta — myös tilastollisten menetelmien käytöstä — on keskittynyt englannin kieleen ja käyttää kieliriippuvia työkaluja ja resursseja, kuten sanaluokittimia ja ontologioita, joita ei voi suoraan soveltaa muihin kieliin. Menetelmät, jotka on kehitetty pelkästään englannille, eivät välttämättä ollenkaan sovi kielille, joissa on erilainen lauserakenne tai kirjoitusjärjestelmä.

Tässä väitöskirjassa tutkitaan ja kehitetään kieliriippumattomia menetelmiä luonnollisen kielen käsittelyyn. Kieliriippumattomia menetelmiä voidaan soveltaa useisiin kieliin ilman tarvetta ylimääräisille kielikohtaisille esikäsittelyvaiheille. Myös murteita, kielten historiallisia muotoja, pieniä kieliä ja erityisalojen kieltä voidaan käsitellä kieliriippumattomilla menetelmillä.

Yksi tämän väitöskirjan keskeinen tulos on kieliriippumattoman *Likey*-menetelmän kehittäminen ja soveltaminen avainfraasien hakuun ja piirrevalintaan. Menetelmää on sovellettu avainfraasien hakuun tietosanakirja- ja tieteellisistä artikkeleista yhdellätoista kielellä ja lisäksi käytetty piirreirrotusmenetelmänä automaattisessa taksonomian oppimisjärjestelmässä sekä uudessa lähestymistavassa käyttäjämallinnukseen dokumenttien vaikeustason analysoinnissa. Toinen väitöskirjan keskeinen tulos liittyy dokumenttien mallinnukseen: työssä on vertailtu dimensionpudotusmenetelmiä ja etäisyysmittoja dokumenttiklusterointitehtävässä, kehitetty uusi kieliriippumaton suora evaluointimenetelmä dokumenttien esitysmuodoille ja käytetty kielitieteellisiä piirteitä dokumenttien klusteroinnissa sanavalintatehtävää varten.

Avainsanat luonnollisen kielen käsittely, laskennallinen kielitiede, ohjaamaton koneoppiminen, kieliriippumattomuus, kielen subjektiivinen käyttö, avainfraasihaku, dokumenttien klusterointi

ISBN (painettu) 978-952-60-4833-8 **ISBN (pdf)** 978-952-60-4834-5

ISSN-L 1799-4934 **ISSN (painettu)** 1799-4934 **ISSN (pdf)** 1799-4942

Julkaisupaikka Espoo **Painopaikka** Helsinki **Vuosi** 2012

Sivumäärä 250 **urn** http://urn.fi/URN:ISBN:978-952-60-4834-5

PREFACE

What a journey it has been! Six and half years of joy and grief, wild grins and hopeless sobbing, bizarre surprise movements in accidental directions, hard hard work (never mention doctoral students and no holidays again!) and occasional frivolous laziness. Once it's over it's all such a relief, but at the same time, it made me addicted to pedantic punctuation rules, late mornings, wordplays, the many 'one more cups' of coffee and the weird attitude of feeling 'just ok' about sometimes working through the night – especially just before a deadline. During these years, life has shown me many facts about reality, both positive and unpleasant, and I have to say that without those I wouldn't be as ready as I feel now: ready to change track and have another set of bizarre movements to totally new directions.

The work for this dissertation was done in the Aalto University School of Science, at the Department of Information and Computer Science and its ancestors the Helsinki University of Technology and the Laboratory of Computer and Information Science. The work was funded by Academy of Finland through the Adaptive Informatics Research Centre (AIRC) Centre of Excellence, the Finnish Graduate School in Language Studies (Langnet), the Finnish Funding Agency for Technology and Innovation (Tekes) through Content Factory project, and the Finnish Foundation for Technology Promotion (TES). I appreciate the financial support – it made this dissertation possible.

While this really is a personal achievement for me, I could never have it done without the help and also just the presence of so many of you. First of all, I am grateful to my supervisor Prof. Erkki Oja for the iron strong professionalism in every detail of the academic work, which I had the privilege to enjoy during the time of my doctoral studies. I always felt that you completely concentrated on the various problems I happened to have at hand, and showed how to handle them with your own example.

I am grateful to my instructor Doc. Timo Honkela for making it possible to study this field of science that has become so dear to me. Thank you also for the possibility to learn so many new things beyond the mere research – these are today and will be during the next years some of the most valuable things for me work-wise. Thank you to my second instructor Dr. Mathias Creutz for structuring the thesis work at its last stage, and for your great help to finalize the thesis. You were also the first person to introduce me to the academic world and its peculiarities. Thanks for all the fun!

I am very grateful to the pre-examiners of the dissertation, Dr. Reinhard Rapp and Dr. Roman Yangarber for your most valuable and detailed comments which made this work a much better one. Also it was such a wonderful feeling to notice for the first time in my academic career that the reviewer has really read the whole text I've written!

My colleagues in the Computational Cognitive Systems research group have been invaluable: on one hand in understanding more about machine learning, language and scientific writing; and on the other hand, the real friendliness, help and support I've enjoyed in your midst. Thanks Sami for being such a great office mate; you tolerated my concerns – whether being details of a method or just distress about life. I owe you so much. Thanks Sami, Oskar and Jaakko for your helpful attitude in spreading your knowledge on all the stuff related to machine learning and natural language processing. Thanks Ilari for all the technical help and nice discussions. Thanks Tiina for your mental support and for being a peer. Thanks Juha for being so understanding and feet-on-the-ground at the same time. Thanks Krista for teaching how a journal article really should be written. Thanks Paul, Mikaela, Tommi and Eric for studying together with me. Thanks my co-authors Alberto, Ilkka, Matti, Antti and Marja for the possibility to write together. It was just so cool to collaborate with you! I also collaborated within two

Tekes projects with several people from other research centres, which I see very valuable. Thanks especially to Tanja for teaching me about research in a non-engineering field and for the innovative and meeting-full time with all the case studies and other things we did together!

My everyday work-life during my doctoral studies was located mostly in the Information and Computer Science department in Otaniemi. The people working there have made it a very pleasant workplace. A nice comment and a smile when passing by, a friendly question about how's it going, an understanding listener at a moment of despair. At this very moment I may not remember you all during the years but please forgive me: you've noticed the delight and happiness you were able to raise and the smile you managed to summon on my face. My sincere thanks especially to Antti S., Dušan, Francesco, Ilkka H., Jaakko T., Jussi G., Kristian, Momo, Nicolau, Ricardo, Ulpu and Yoan for discussions and all the fun. Thanks Leila, Tarja and Minna for all the patience and help you provided. Special thanks to Prof. Olli Simula for being a great leader of a research unit of any size. I could trust that you make the decisions in the best possible way, also from the doctoral student point of view. Many thanks to the artistic group who helped me with the cover picture: Giorgi, Hongyu, Nima, Paula, Ricardo, Ritabrata, Shinnosuke and Yoan. Finally, thanks Sumu V. for taking care of all the tasks I didn't want to do, for taking the blame for problems and failures, and for sending all those nice notes and presents!

And when I stopped working after a brutal day, there were those people who kept my mind more balanced by strongly emphasizing everything that is not linguistic and especially not machine learning. With you, I have felt relaxed and happy, had time to do various things that interest me, and obtained plenty of fresh new energy for the next work day. Thanks Henna, Erkka and Rio for the help and support when badly in need and joining the fun on happy days. Thanks Heikki for the wonderful time. Thanks Meri, Taru, Päpä and Eve for all the play and fun. Thanks Ville I., Suvi, Pauli and Ville T. for the pleasant lunch breaks. Thanks Ann for checking this text and everything else.

Thanks my mother, Arja, and father, Raimo for everything you've done for me, your endless trust in me and my skills in studying and work. Thanks my grandmother Eila and grandfather Pauli for showing me what it is when you have collected life experience. Thanks my brothers Rauno and Riku and sister Sara for being just you – that's much more than I could hope. Thanks my sister Erkku for everything. You are irreplaceable.

In Espoo, Finland, 13th October 2012

Mari-Sanna Paukkeri

CONTENTS

LIST OF PUBLICATIONS

This thesis consists of an introductory part and the following publications, which are referred to by their Roman numerals.

I Mari-Sanna Paukkeri, Ilari T. Nieminen, Matti Pöllä, and Timo Honkela, 2008. A Language-Independent Approach to Keyphrase Extraction and Evaluation. In *Coling 2008: Companion volume: Posters*, pages 83–86, Manchester, UK. Association for Computational Linguistics.

II Mari-Sanna Paukkeri and Timo Honkela, 2010. Likey: Unsupervised Language-Independent Keyphrase Extraction. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*, pages 162–165, Uppsala, Sweden. Association for Computational Linguistics.

III Mari-Sanna Paukkeri, Alberto Pérez García-Plaza, Víctor Fresno, Raquel Martínez Unanue, and Timo Honkela, 2012. Learning a taxonomy from a set of text documents. *Applied Soft Computing*, 12(3):1138–1148. Elsevier B.V.

IV Mari-Sanna Paukkeri, Jaakko Väyrynen, and Antti Arppe, 2012. Exploring Extensive Linguistic Feature Sets in Near-Synonym Lexical Choice. In A. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing, 13th International Conference, CICLing 2012, New Delhi, India, March 11–17, 2012, Proceedings, Part II*, volume 7182 of *Lecture Notes in Computer Science*, pages 1–12. Springer-Verlag, Berlin/Heidelberg, Germany.

V Mari-Sanna Paukkeri, Ilkka Kivimäki, Santosh Tirunagari, Erkki Oja, and Timo Honkela, 2011. Effect of Dimensionality Reduction on Different Distance Measures in Document Clustering. In B. L. Lu, L. Zhang, and J. T. Kwok, editors, *Neural Information Processing, 18th International Conference, ICONIP 2011, Shanghai, China, November 13–17, 2011, Proceedings, Part III*, volume 7064 of *Lecture Notes in Computer Science*, pages 167–176. Springer-Verlag, Berlin/Heidelberg, Germany.

VI Sami Virpioja, Mari-Sanna Paukkeri, Abhishek Tripathi, Tiina Lindh-Knuutila, and Krista Lagus, 2012. Evaluating vector space models with canonical correlation analysis. *Natural Language Engineering*, 18(3):399–436. Cambridge University Press.

VII Mari-Sanna Paukkeri, Marja Ollikainen and Timo Honkela, in print. Assessing user-specific difficulty of documents. Accepted for publication in *Information Processing and Management*. Elsevier Ltd.

ACRONYMS

ANN	Artificial neural network
ARI	Automated readability index
CCA	Canonical correlation analysis
EM	Expectation maximization
F	F-measure
HMM	Hidden Markov model
ICA	Independent component analysis
IE	Information extraction
IR	Information retrieval
kNN	k nearest neighbours
LDA	Latent Dirichlet allocation
LSA	Latent semantic analysis
LSI	Latent semantic indexing
MAP	Mean average precision
ME	Maximum entropy
MI	Mutual information
ML	Machine learning
MLE	Maximum likelihood estimate
MNR	Multinomial logistic regression
MT	Machine translation
NER	Named entity recognition
NLP	Natural language processing
NMI	Normalized mutual information
NB	Naive Bayes
NMF	Non-negative matrix factorization
OOV	Out-of-vocabulary
P	Precision
PCA	Principal component analysis
PLSA	Probabilistic latent semantic analysis
R	Recall
SMT	Statistical machine translation
SNLP	Statistical natural language processing

SOM	Self-organizing map
SVD	Singular value decomposition
SVM	Support vector machine
TP	Global taxonomic precision
TR	Global taxonomic recall
TF	Global taxonomic F-measure
VSM	Vector space model
WSD	Word sense disambiguation
WSI	Word sense induction

Linguistic acronyms

N	Noun
NP	Noun phrase
O	Object
S	Subject
V	Verb

Languages

da	Danish
de	German
el	Greek
en	English
es	Spanish
fi	Finnish
fr	French
it	Italian
nl	Dutch
pt	Portuguese
sv	Swedish

t_i	Term (word or phrase)
n	Phrase length
$c_j(t)$	Term frequency (count of term t in document j)
$d(t)$	Document frequency (number of documents containing term t)
$G(t)$	Global frequency of term t in a document collection
$\mathcal{X}$	A discrete set of symbols
m_i	Number of observations
M	Total number of samples
X, Y, x	Random variables
μ	Mean
σ^2	Variance
$\mathcal{N}$	Gaussian distribution
$p(x)$	Probability of x
$E(X)$	Expectation
$H(X)$	Entropy
$H(X,Y)$	Joint entropy
$I(X;Y)$	Mutual information
D	Target dimensionality
$\mathbf{a}$, $\mathbf{b}$, $\mathbf{x}$, $\mathbf{y}$	Data vectors
$\mathbf{X}$, $\mathbf{Y}$	Data matrices
$\mathbf{X}_D$	Data matrix in new dimensionality D
$\mathbf{U}$, $\mathbf{V}$	Unitary matrices
Σ	Diagonal matrix
$\mathbf{W}$, $\mathbf{H}$	Weight matrices
$\mathbf{W}_D$	Weight matrix of dimensionality D
$\mathbf{S}$, $\mathbf{T}$	Document collections
$\mathbf{S}^{\mathrm{T}}$	Transpose of matrix $\mathbf{S}$
$\mathbf{z}$	Language-independent semantic space

k	Number of clusters
$\mathbf{c}_k$	Centroid of cluster k
$S_k \subset S$	Subset of a data set
ρ	Correlation
u_1, v_1	Scalars
$L(t,j)$	*Likey* score for term t in document j
$w(t,j)$	*Likey* weight for term t in document j
a	$Likey_N$ constant
η	$Likey_N$ threshold
τ	$Likey_W$ parameter
φ	$Likey_P$ parameter (number of keyphrases per 100 words)
ξ	*Likey* post-processing threshold
$\mathcal{T}$	Document feature space
$\mathbf{u}_k$	User vector
$\mathbf{d}_j$	Document vector
$\delta(\mathbf{d},\mathbf{u})$	Difficulty of document $\mathbf{d}$ for user $\mathbf{u}$
$\theta(t)$	Perceived difficulty of term t by humans
M_j	Length of document j
N	Number of documents in a collection
U	Number of users

sun	the concept of sun
'sun'	the lexical form of the concept *sun*

1

INTRODUCTION

Natural language processing (NLP) is the field of research for automatic processing of natural languages, such as English, Finnish, or Ancient Greek. The recent very quick development of technology has increased the amount of text collections available for everyone, and this, rather new data can be used for many purposes that have not been possible earlier. An example of the new possibilities is statistical machine translation which requires a large number of translated texts to construct a translation model. Electronic documents in different domains, such as information technology or medicine, can be easily collected, and the used language and vocabulary compared and analyzed both qualitatively and quantitatively. Online discussions on topics such as products, services or hobbies provide data about the needs, opinions and behaviour of consumers and temporal variations in them. The discussions represent the actual usage of languages, already in electronic form and in huge amounts. For example, behavioral scientists and companies, as well as linguists and computational linguists, can obtain totally new kind of information about people. The major source of information in electronic form is the Web, in which an increasing amount of information is freely available. In addition, individuals and societies can have their own repositories of documents and reports, and companies collect huge databases of product development, customers, markets, competitors, etc.

Many of us are happy that the Google search engine works also for the Finnish language[1], as well as for many other languages. Likewise, it is convenient that commercial text processing software comes with the special characters, a spellchecker and a hyphenator of our own language. Today, practically any area of research and any area of industry benefits from automatic text processing. The examples show very well that we, the citizens of the 21st century, already have an assumption of the availability of this 'basic' language processing software and that we further assume that the systems work also with our own languages. However, the widely known software and services are only the tip of the iceberg in the various possibilities provided by natural language processing and text mining techniques.

To be able to analyze data, interesting information is collected into statistics, in a *structured* format, a format easy to access. However, many of the statistics are collected manually from *unstructured* text or other unstructured formats. Traditionally, both scientific and commercial research studies have been restricted with the reading capacity of the researchers, since every piece of information has been picked, stored and analysed manually. To be able to utilize information in large databases of unstructured textual data, some NLP applications are needed, such as search from the database and automatic summarization of the search results. Sophisticated text data analysis methods together

1 `http://www.google.fi`

with the increasing amount of data provide a possibility of much wider and deeper data analysis and knowledge extraction than ever before.

This thesis discusses automatic language-independent methods for natural language processing. The thread running through the thesis consists of three main themes: *fully automatic approaches, language independence* and *subjectivity of language use*. All of these will be considered from the viewpoint of computer-based data processing, and, at the same time, from the viewpoint of processing natural languages as the data. The themes have been developed in the Computational cognitive systems research group led by Timo Honkela (Honkela *et al.*, 2008, 2010; Lagus *et al.*, 1999). In this introductory part of the thesis, the themes are further developed, and their connections and influence to the approaches introduced in this thesis are shown. Especially the introductory part proposes a definition for language independence in natural language processing and discusses the related literature. The thesis contributes to the field of computational linguistic which is a multidisciplinary field, by combining linguistic research with machine learning. In the introductory part, background information is sometimes given on a rather elementary level: The goal is to give sufficient information for both computer scientists about linguistics and linguists about machine learning to understand the multidisciplinary publications.

FULLY AUTOMATIC APPROACHES In machine learning of language data, and especially in the unsupervised machine learning domain, fully automatic processing of data is a natural choice. Fully automatic approaches do not use manual intervention by labelling data items manually, selecting parameters or constructing rules, etc. In many other areas, such as (especially traditional) computational linguistics, information theory, robotics, fuzzy systems, etc., it is not unusual to manually create rules for a system. A non-automatic approach makes a hypothesis about what kind of structure is to be found. For small data sets, it would perhaps be the most effective procedure, but in the case of large data sets, manual analysis is time-consuming and difficult. Statistical and data mining methods can find relations and structures that the researcher did not even think about. This data-driven approach is typical to data mining methods: the goal is not to validate a hypothesis with the data but to use the data to automatically construct a model about a phenomenon.

LANGUAGE INDEPENDENCE Within the NLP community, there has been a lot of effort to analyse individual languages separately. There are thousands of languages in the world, in addition to all the dialects which usually also need separate text processing. If every text processing algorithm, including methods in information retrieval, machine translation, information extraction and so on, had to be developed separately for each of the languages, the number of different systems and the amount of work required would be gigantic. Language-independent methods are a solution to the problem: the goal is to formulate automatic procedures that apply to languages in general, not to a specific language only. Further advantages of language independence are the ability to process languages of few speakers, or ancient versions or dialects of languages, for

which language-specific resources do not exist. The possibility to port a system to languages of an expert or other specific domain is another advantage. More specifically, an international company could handle documents in all of its official languages with reasonable costs if language-independent methods were used. A language-independent method is not the same as a fully automatic method. For example, some methods highly independent of the used language use few seed items that have to be collected manually and thus are not fully automatic. On the other hand, there are fully automatic methods that work for a single language only, because of the use of a language-specific resource.

SUBJECTIVITY OF LANGUAGE USE Even if two people are native speakers of the same language, they have subjective viewpoints, subjective world knowledge, subjective interests, and their own subjective versions of the language. Human beings are very flexible to understand what other people mean, even though they would not use the same words and phrases themselves. Subjectivity of language use can be easily demonstrated with a test in which a non-prototypical colour is shown to several people and they are asked to name it. Typically, many different names are given for the same colour. In addition to having many names for the same entity and subjective preferences for using the names, also the meanings and associations related to words vary: for example a 'good person' may have a meaning of 'empathetic' to one, 'donator to charity' to another, and 'diligent' to a third person. Furthermore, the pragmatic preferences for selecting an utterance in each situation vary a lot. All of these aspects relating to the subjectivity of language use have to be taken into account if the purpose is that computers adapt to the language used by people, rather than people adapting to the (very simple) language used by computers. Statistical methods, in contrast to their rule-based counterparts, can take this kind of subjective variations into account, but subjectivity has not, however, been studied much yet.

1.1 SCOPE OF THE DISSERTATION

In this dissertation, fully automatic methods, language independence and subjectivity are considered in several natural language processing tasks. A fully automatic and language-independent approach for *keyphrase extraction* is presented and its performance is shown for 11 European languages, including English and Finnish (Publications I, II). Keyphrase extraction is an NLP task in which the semantically most interesting terms, keyphrases, are extracted from text documents. The research on keyphrase extraction has been growing during the last five years but it has mostly been concentrated on documents in English. Also a language-independent evaluation method for keyphrase extraction is presented (Publication I).

If the goal is to develop language-independent, fully automatic and subjective methods, the use of ontologies is not very tempting, because of their assumption of objectivity in how concepts are related, their language specificity and the typical way of constructing ontologies manually. An alternative approach is to use machine learning techniques that do not need manual anno-

tations or manually defined rules. Nevertheless, the use of ontologies has the advantages of fairly accurate information (and, even, the fact that there exists *some* information) that would be difficult to obtain with pure machine learning. In this thesis, an approach for *learning taxonomies* from encyclopedia documents is proposed (Publication III). The work is an early step to automate the construction of ontologies and get ontologies more applicable to a multilingual domain.

Word sense disambiguation is a well-known supervised NLP task. In a related problem, *lexical choice*, machine learning methods are applied to a collection of as many linguistic features as possible to study how the linguistic features help in the machine learning task (Publication IV). Because of the linguistic annotation, this approach is not language-independent and takes an opposite approach to the other studies. However, part of the methods applied are unsupervised and thus follow the viewpoint taken in this dissertation.

All of the methods mentioned above have a *feature extraction* step: a step in which the important features for the task are collected from text documents and the rest of the information is discarded. The feature extraction step is studied by analysing the effect of different dimensionality reduction, normalization and distance measures in the task of document clustering (Publication V) and proposing an evaluation method for feature extraction (or document representation) (Publication VI). To further show the level of language independence of these methods, the experiments are run with several languages from different language families.

The third main theme, subjectivity of language use, is specifically considered in a task of assessing the difficulty of a text. A novel approach is proposed, in which the difficulty assessment is done separately for each user (Publication VII). In contrast to the traditional readability measures for difficulty assessment, the proposed method is intended for assessing suitable documents for adults that have knowledge of varying expertise areas.

1.2 SCIENTIFIC CONTRIBUTIONS OF THE PUBLICATIONS AND AUTHOR'S CONTRIBUTION IN THEM

Publication I proposes a novel keyphrase extraction method, *Likey*. In contrast to most of the other methods, the proposed method is language-independent and directly applicable to a new language, provided that a reference corpus is available. The experiments are run for Wikipedia articles in 11 European languages. As another major contribution, the paper proposes a language-independent evaluation method based on Wikipedia interlinkings. The author of the thesis is responsible for the article setting, further development of the original idea (the use of relative ranks), the evaluation method, part of the implementation and most of the writing.

In **Publication II**, the *Likey* method is applied to English scientific articles. Also a modification to the post-processing is proposed. The performance of the method is compared to two reference keyphrase sets and three baseline methods. The author is responsible for the whole study: the idea, implementation, experiments and writing the article.

Publication III applies the *Likey* method, together with two other feature extraction methods, to taxonomy learning. The method is novel as it uses the text contents of Wikipedia articles as concept definitions and clusters them hierarchically to obtain the taxonomy. Moreover, the method with *Likey* feature extraction is language-independent. The experiments are run for three languages: Finnish, English and Spanish. The author is responsible for most of the details of the method (outside the fuzzy logic part), part of the implementation, and part of the experiments. The author and the second author of the journal article did most of the writing, while the author had the main responsibility.

In **Publication IV**, a new data set in a new language, Finnish, is proposed for a lexical choice task. Different machine learning paradigms, unsupervised, semi-supervised and supervised, are compared using an extensive set of linguistic features, containing semantic, syntactic and morphological features, far more than present in previous work. The results show that although purely syntactic linguistic features play the biggest role in performance, also semantic and morphological features are needed. The author is partly responsible for the experimental setup and experiments, and wrote the paper together with the second author but having the main responsibility.

In **Publication V**, three dimensionality reduction methods and 10 distance measures for document clustering with *k-means* are tested. The effect of dimensionality reduction into a range of target dimensionalities for different distance measures are analysed using toy data. The experiments are run for English and Hindi and they show that the methods commonly used in document clustering do not always give the best performance. The author is partly responsible for the idea, the experimental setup, and the experiments and did most of the writing of the paper.

A novel language-independent direct evaluation method for document representations based on canonical correlation analysis is proposed in **Publication VI** to fill the gap in the language-independent direct evaluation methods. The method is analysed thoroughly and its performance compared to known facts about feature extraction in the literature, sentence-matching task and manual validation. The experiments are run for bilingual combinations of English, Finnish, Danish, German and Swedish. The author participated in designing the experiments and writing the article. The author is responsible for the manual validation and part of the related work.

A further application of the *Likey* method is proposed in **Publication VII**. The article proposes a novel method for text difficulty assessment. In contrast to earlier methods, the difficulty measure is user-specific and can thus handle the variation of expertise levels on different domains each user has. The measure combines user and document vectors which have been created using *Likey*. The experiments are run for the Finnish language in the medical domain. The author participated in developing the original idea, and is responsible for the proposed difficulty measure. The author is partly responsible for the experimental setup of the first method and fully responsible for the second method. The author also ran most of the experiments (excluding the human evaluation part) and did most of the writing.

1.3 STRUCTURE OF THE INTRODUCTORY PART

The introductory part of this dissertation is organized as follows. Chapter 2 goes through characteristics of natural language from a machine point of view. Different uses of written language are covered and the ways how a machine processes written language are discussed. The meaning of words, phrases, sentences and text passages are of uttermost importance in order to be able to do intelligent text processing. Semantics and related topics are discussed in Section 2.2. Quite recently, computational linguistics research has started to concentrate more on also other languages than English. Differences between languages make the extension of methods developed originally for English to other languages very complicated. In order to understand the difficulties arising from multilinguality, this topic is discussed in Section 2.3.

Language-independent approaches for natural language processing are discussed in Chapter 3. The concept of language independence is introduced in Section 3.1 and the chapter continues with machine learning paradigms, the possibilities in language-independent preprocessing, and a review of unsupervised machine learning methods.

NLP research consists of a large collection of different problems and tasks. In Chapter 4, the new approaches and methods developed within this dissertation are presented in the context of other language-independent approaches in the literature. A language-independent approach to the task of keyphrase extraction is presented in Section 4.1 and further applied to the task of taxonomy learning in Section 4.2. In the lexical choice task in Section 4.3, a range of clustering methods are tested on a large set of linguistic features. In Section 4.4, two topics in creating document representations are discussed. The first topic is document clustering with the *k-means* method and the second topic is the evaluation of document representations. One of the main themes of this dissertation, subjectivity, is considered in Section 4.5. An approach for user modelling in text difficulty assessment is presented. In Section 4.6, other NLP tasks that can be approached with language-independent methods, information retrieval and machine translation, are discussed.

Chapter 5 concludes the work and gives some future views of language independence in natural language processing in a multilingual domain.

2

NATURAL LANGUAGE IN TEXT PROCESSING SYSTEMS

Natural language processing (NLP) is a research field that combines natural languages and linguistic knowledge with computer science and machine learning. *Computational linguistics* is another name for natural language processing. The field studies language as data and aims to develop sophisticated methods and applications for humans to use, such as systems for information retrieval, text classification and machine translation. *Statistical natural language processing* (SNLP) uses statistical methods for NLP, as an alternative to rule-based approaches. *Text mining* means somewhat the same as SNLP, but it comes from the tradition of *data mining*, the development of machine learning methods for large data sets, applied to text data. This thesis concentrates on language in textual form, whether it has been transcribed from speech or been originally written language. The main difference between written and spoken (recorded) language from a computer point of view is that in text data the units (e.g., letters or characters) are distinguishable from each other and clear, whereas speech data is more ambiguous and noisy.

This chapter discusses the characteristics of written natural language from the automatic text processing point of view: What kind of features have to be taken into account to be able to use automatic methods to analyse natural language? It is also discussed how natural language differs from other kinds of data in the field of data mining; what advantages and shortcomings natural language has compared to other data. Next, a short introduction to semantics, the meanings of words and sentences, is given, with a discussion how the complex problem of understanding context-dependent variations in the meaning is approached in computational linguistics. Many of us speak and write also in other languages than English, and thus multilinguality is considered next, aiming to get an understanding what kind of differences and similarities there are between languages. Even though this thesis is concentrated on language-independent NLP methods, the automatic methods discussed in this chapter are not necessarily language-independent. However, they give an interesting overview on the various possibilities in analysing language and measuring different characteristics of it.

2.1 NATURAL LANGUAGE

Language has a big role in our culture and everyday lives. A visit to a country in which people speak and write a language that you do not understand shows how important it is to have a common language with the people around you. Language, the ability to speak, write and understand others, has a multitude of functions for a human being. Finch (2003) listed different functions, part of which go far beyond the obvious 'giving information' and 'reading information':

- *Physiological function (no communicative purpose)* to release nervous or physical energy
- *Phatic function* for purposes of sociability: greetings, small talk, ...
- *Recording function* to provide a record: shopping list, diary, ...
- *Identifying function* to identify and classify things: names, terminology, ...
- *Reasoning function* as an instrument of thought
- *Communicating function* as a means of communicating ideas and feelings
- *Pleasure function* to give delight: poems, [jokes], ...

Language can be said to be pervasive in human life. Different functions produce also different ways to use words, varying from a list of separate words, via the use of conventionalized phrases such as greetings and small talk, to artistic approaches playing with rhymes and graphical text.

Speech and conversation can be seen as the original form of natural language. The writing systems have been invented much later. The written form is very important in storing information and communication in large chunks, such as letters or books, and in studying. Written language and the research of it can be divided into a set of subsystems. The *vocabulary* of a language contains words that are part of the language. The vocabulary is neither static nor easy to define because words are repeatedly borrowed from other languages and totally new words are invented regularly. Similarly, words that are not needed anymore are forgotten from a language. Another subsystem is *semantics*, the meanings of words and sentences. To be able to put words together to construct understandable and 'correct' sentences, knowledge of *syntax* or *grammar* is needed. *Morphology* analyses meaningful linguistic units, such as word stems, affixes and suffixes. *Pragmatics* studies the actual use of language in practice, beyond its literal meaning. Spoken language is in focus on fields like *phonology* and *phonetics*, but they are not discussed here.

Even though languages are often considered to be uniform systems, they do not stay the same during the decades and centuries. In addition to the changes in the vocabulary, also the meanings of individual words and grammatical structures change. Within a single language, considerable differences exist between different geographic areas or social classes. Old text, such as Shakespeare's plays from late 16th century and early 17th, and the Kalevala of Finnish and Karelian folklore from the 19th century, are nice examples how different language, at least the written form, has been only a few centuries back, compared to the contemporary usage.

2.1.1 *Automatic approach and word context*

The variations in the use of natural languages and their characteristics are endless but when looking at language as data given for a computer to analyse, it

is simply a sequence of symbols. Text can be seen as a discrete-valued one-dimensional vector, having a finite number of different symbols (letters or characters). The occurrence of a word is highly dependent on the previous words in a text. However, roughly almost a half of word types occur only once in a data set of any size, depending slightly on the language. The most common words in each language occur in almost every sentence, such as 'the' or 'and'.

NLP and especially statistical NLP methods are based on the word *context*. Words and their co-occurrences with other words are almost the only way to obtain information about the meaning and use of a word if no additional sources of information are used. Because of the large number of words that occur only once in a piece of text, language is usually analysed in larger chunks than few consecutive words: in sentences, paragraphs or whole documents. These contexts are applied by counting the frequencies of all the context words. In spite of the fact that many natural language words are ambiguous, such as 'set' or 'light', many NLP applications make simplifying assumptions that each word has only one sense per discourse (Gale *et al.*, 1992) or per collocation, that is, one sense per each combination with other words (Yarowsky, 1993). Martinez and Agirre (2000) showed that the collocations actually vary from one corpus to another, depending on the topic and genre.

2.1.2 *Topic, domain and genre*

A piece of text typically has a *topic* it talks about. A topic may be for example a concept, such as *education* or *cooking*, or a phrase or a short sentence 'the presidents of the USA' or 'singing together'. *Topical words* in a document can be found automatically by using, for example, keyphrase extraction methods (Frank *et al.*, 1999; Publication I). *Topic modelling* is a statistical approach for automatically finding out the topic or a set of topics in a text by using the distribution of the words and their frequencies.

Domain is another, and somewhat wider, concept than topic and it also has a big role in the vocabulary of a text. Some examples of different domains are *medicine*, *technology* and *housing construction*. *General domain*, or domain-neutral, texts do not contain domain-specific vocabulary or jargon. Many NLP applications have been developed for a *specific domain*, because they use more restricted language. A good example domain is the weather forecast texts. NLP applications developed for a specific domain do not easily extend to domains outside the case, because outside the specific domain, known terms have new meanings and new terminology appears (Agirre *et al.*, 2009). In the case of new terms, these *out-of-vocabulary* (OOV) words cannot be recognized with traditional NLP tools. In a new domain, also the contexts of words change and thus the calculated word distributions do not hold anymore. Escudero *et al.* (2000) reported that their systems trained on general-domain texts performed worse when applied to a specific domain of economics.

To be able to successfully apply an NLP application to a new domain, methods for *domain adaptation* from general domain to a special domain have been proposed. The approaches require text collections from both general and the specific domain, either labelled (tagged) (Agirre *et al.*, 2009) or unlabelled

(Blitzer *et al.*, 2006). Also domain adaptation between two specific domains has been studied, using additional general domain data (Daumé III, 2007; Daumé III *et al.*, 2010).

Besides the domain and the topic, written language can be categorised according to the document *genre*, such as *newspaper article*, *treatment recommendation*, *letter*, or *obituary*. Genres of online materials can be, e.g., *personal homepage*, *public homepage*, or *frequently asked questions* (FAQ's) (Lim *et al.*, 2005). Through the invasion of the Internet, *dynamic documents* have become a new genre. Some genres involve highly literary language, whereas some others are more casual, using rules of colloquial language, or even ignoring part of the grammatical or spelling rules and inventing their own vocabulary. Even inside a domain, for example, a research field, there exists a large variety of genres. Over fifty genres of medical texts were identified (Zweigenbaum *et al.*, 2001) while constructing a representative corpus of French medical language. The genres included for example reports, letters, teaching and reference materials, publications, guidelines and official documents. The genre of a document can be detected or classified automatically using statistics on, for example, part-of-speech tags, word lengths, and punctuations (Lim *et al.*, 2005; Finn and Kushmerick, 2006; Lijffijt *et al.*, 2011).

The genre directs the used vocabulary, phrases, style, and topic of a text. From the viewpoint of automatic methods the shifts between genres are difficult because of the changes in vocabulary, word frequency distributions and co-occurrences. For example, a machine translation system trained on newspaper articles may not be very efficient in translating letters or personal homepages.

2.1.3 *Subjective use of language*

While there are many genres that people have to, or which they want to follow when they write, every native speaker has an individual version of the language to use, so-called *idiolect*. People have a very large active vocabulary and an even larger passive vocabulary. Both the words and their usage preferences vary from time to time and from speaker to speaker in both the vocabularies. In a study of spontaneous word choice for objects, two people favoured the same term with the probability of less than 20% (Furnas *et al.*, 1987). Even one individual uses different words at different times for saying the same thought (Coulthard, 2004), which can be partly explained with the *priming* effect of short-term memory (Farrell *et al.*, 2012). A simple example of this is the greetings: yesterday I said 'hello' to you, today it was 'hi', and tomorrow it may be 'Oh, how tired I feel'.

Finch (2003, p. 212) listed different styles of writing, which illustrates the possible variations between people, and also the possibilities an individual can choose from:

- *Phrases or sentences:* incomplete or complete; one or more clauses
- *Constructions:* active, passive, transitive or intransitive; tense preferences; lexical, auxiliary or modal
- *Register:* formal, technical or slang; polysemy; use of figurative language

- *Mood:* interrogative, imperative or declarative
- *Graphical text:* font, font size, spacing

In addition to these, humorous style can be used for example in novels or newspaper columns. Furthermore, there are differences in the compactness of writing: someone writes very precisely and briefly, whereas another person is rather elaborate. Also systematic use of ungrammatical language (such as spelling errors) or unconventional ('incorrect') use of vocabulary are personal characteristics which may stay across domains and genres. Reiter and Sripada (2004) studied near-synonym choices in English weather forecast texts and suggested a set of factors that are important when the author chooses the actual wording: preferences and idiosyncrasies of individual authors, collocations, the stylistic requirement of variation in the lexical usage, and position of a lexeme in a text.

Besides having their own preferences for certain words and structures, people have their own unique interpretation for each word, different from those by other people. This is called the *indeterminacy* of linguistic meaning: the impossibility of determining absolutely what a person has actually meant with an utterance (Finch, 2003, p. 130). Luckily, a human being is very adaptive in many ways, and also in understanding language.

Authorship characterization is a task that predicts characteristics of a text author, such as the gender (Kucukyilmaz *et al.*, 2006), or educational or cultural background. Thomson and Murachver (2001) showed that the differences between genders were small in individual linguistic features, but when combining them, it was possible to predict the correct gender by 88–95% accuracy. A further task is *authorship identification* in which the goal is to recognize an individual author. Possible application areas are, e.g., plagiarism detection and forensic linguistics (Coulthard, 2004). There are many features in the text that have been proposed to be used for automatic identification of the author: distributions of characters and frequencies of function words (Juola and Baayen, 2005), net abbreviation, orthographic and placement features (Argamon *et al.*, 2003), keywords (Zheng *et al.*, 2006), and features about signatures and attachments in e-mails (de Vel *et al.*, 2001). Li *et al.* (2006) selected an optimal feature set automatically for English and Chinese online messages and achieved identification accuracy of 99% and 94%, respectively, within a group of 10 authors.

NLP methods often dismiss the subjectivity of language use discussed above. An example is the extensive use of ontologies and semantic categorizations which do not allow subjective variations. Although the assumption of objectivity is applicable in some tasks, such as syntactic analysis, many systems apply so-called objective semantics that would easily be a wrong assumption in other domains and similarly wrong with other (group of) people. A wrong assumption in a specific domain would be, e.g., the meaning of word 'well' which is common in almost every domain, but has a specific meaning when talking about water or oil holes in the ground. The correct interpretation of subjective words such as 'me' and 'you' are natural for human beings but not for a computer. Some other examples about subjective semantics are the word 'domestic', which has different meaning for people from different countries, and

names like 'John', which points to different individuals according to the Johns known by the speaker.

User-specific NLP methods pay attention to the unique and individual language written and spoken by the user. An example is Publication VII, in which document difficulty is measured for each user individually, not on the basis of an (imaginary) average user. Also many existing statistical NLP approaches, in contrast to rule-based methods, can be seen as dealing with subjective language. Fully statistical approaches usually analyse the language blindly, without preconceptions about, for example, the used words and their meanings. These kinds of methods can also deal with new conventions in language use, misspellings, and non-standard usage such as in blog texts (Karlgren *et al.*, 2012). On the contrary, rule-based methods force the users to use only words on a predefined list of 'accepted' words for which a set of rules can be applied – the rest of the words cannot be analysed, as can be seen in the results of for example rule-based machine translation (Toma, 1977) or lemmatizer (Kanis and Müller, 2005).

The subjectivity of language use is close to the idea of *connectionism* (Macdonald and Macdonald, 1995), a field of cognitive science that incorporates a biologically plausible model of cognition: cognition resembles the brain using neurons with a degree of activation and weighted connections between the neurons. The model assumes that each brain individually learns from the input it gets and builds a model of it – ending up with a different model than any other individual has.

Subjective understanding of text

In the same way as there are huge differences between the authors of text, also the recipients and readers vary a lot. In communication, the messages are intended for a certain audience (Veivo and Huttunen, 1999, p. 101; Mustajoki, 2008). A reader of a piece of text may feel that the text is difficult to understand. Some of the most common reasons for this are the unfamiliarity of the used terminology and the background knowledge required. Crossley *et al.* (2008) listed representation skills the reader needs: world knowledge, knowledge of the text genre and the used discourse model. Further they noted that text comprehensibility includes measures of text cohesion, meaning construction, decoding, syntactic parsing, propositional density, complexity and the amount of working memory (Graesser *et al.*, 2004; Duran *et al.*, 2007; Crossley *et al.*, 2008). Also different kinds of errors or untypical use of language may increase the difficulty of a text: grammatical or spelling errors, words in foreign language, inadequate (e.g., too descriptive) language, ambiguous words and phrases, and structural problems. Content that is new for the reader makes text more difficult (Harley, 1995, p. 209). Some texts are so deeply inside a specific domain that the readers of the texts are required to be experts of the area. For example, the intended target audience *expertise level* of online documents shows a continuous scale (Grabar and Krivine, 2007) between expert and lay documents. Lay people may find texts intended for experts very difficult to read. Also school pupils and people with disabilities, such as dyslexia, have specific needs for the intelligibility of the text, compared to an average adult.

In the literature, a number of techniques have been proposed for the assessment of the difficulty level of a text, ranging from manually calculated readability formulas (e.g., Senter and Smith, 1967; McLaughlin, 1969) to automatic machine learning (Petersen and Ostendorf, 2009; Crossley *et al.*, 2011). However, none of them measures the subjective difficulty according to the skills of a certain user. In this thesis, a user-specific approach for text difficulty assessment is proposed in Publication VII.

2.1.4 *Fully automatic methods*

A *fully automatic* NLP method is an algorithm which can be applied to a new data set without the need of manual intervention. This is not usually the case with NLP methods: the systems typically need parameter tuning, specific preprocessing and even additional resources. A fully automatic method produces the analysis or other result independently, and does not require manual labelling or tagging, translation, selection of features or data items, parameter tuning, or manual pre- or post-processing. These apply to both the stage of system training and the actual use of the system. A system based on statistical learning is closer to fully automatic than a system using symbolic rule-based learning, because the latter requires human expert knowledge in building the rules.

A fully automatic system does not preferably use any hand-encoded knowledge that was constructed specifically for automatic use, such as ontologies or other knowledge bases. However, some NLP systems use manually compiled resources and can still be called fully automatic: These systems apply resources readily available, preferably originally intended for human use, and thus do not require additional manual work to be able to construct the system. Such resources are, for example, parallel corpora or dictionaries. If linguistic resources or other knowledge are readily available for the present task, no further manual work is needed, but in case of transferring the system to another language, and possibly in the case of a changing to a new domain, the manual resource has to be re-constructed.

Fully automatic methods are convenient in multilingual and multi-domain environments because they are easy and quick to train and use. In this dissertation, the goal has been to develop fully automatic methods. The approaches presented in Publications I, VI and VII can be called fully automatic. However, some manual fixes are easy and quick to make, such as determining the appropriate number of clusters for a task, and they may yield significantly better performance of the system. In the rest of the publications, small manual steps have been taken: defining either a threshold value for keyphrase extraction or the number of clusters.

2.2 SEMANTICS

The research field *semantics* studies the meaning that is communicated through language (Saeed, 1997). Philosophers and linguists have been discussing the

meaning of meaning (Ogden and Richards, 1923) and how words and the world are linked to the meaning. Peirce (1931–1958) suggested the notion of *semiotic symbol* in which a *sign* is a relation between a referent, its meaning and form. Steels (2008) renamed the items as *object*, *concept* and *symbol*. In this system, for example, a symbol (lexical form) 'snake' is associated with the concept of *snake*, the idea of what a snake is like and how it relates to other concepts. The object may be concrete, like the snake, or an abstract thing. Vogt (2003) applied the idea of the semiotic symbol to autonomous robots and language games, in which the robots developed a shared lexicon by playing games. This was further developed by Lindh-Knuutila *et al.* (2006) using an artificial neural network as a model of the conceptual memory of an agent. These approaches assume that each agent, similarly to human beings, have a subjective conceptual memory, i.e., the representation of the *meaning* of words, to which words in the used language are mapped. The meanings are then further mapped to instances in the world. The different meanings can be seen from another point of view when considering a discourse between two people: a surface ('open' or shared meaning, one of which all parties can be aware), the meaning by the speaker ('concealed', not consciously known to the listener), the meaning by the listener, and the meaning by an (accidental) hearer ('blind') (Finch, 2003, p. 130).

There are many different relationships between words and meanings. *Homonymy* refers to two or more words that have the same lexical form but the meanings are different. The words were originally from distinct lexical sources but they ended up with the same form. A related term is *polysemy*, which also refers to different meanings of the same surface forms. In this case, the units have been derived from the same lexical source and have been processed by extension, such as *metaphor* or *metonymy* (Croft and Cruse, 2004, p. 111). However, usually there is no need to make this distinction between homonymy and polysemy in automatic processing of language. Other interesting relations are *synonymy*, words with different lexical forms but similar meanings, and *antonymy*, words with opposing meanings. From a categorization point of view, the definition of *hypernym* as the super concept and *hyponym* as a subconcept, are important. Besides the associations listed above, words and concepts may also be associated in human experience. For example, *restaurant* is associated with *customer*, *waiter*, *ordering*, *eating* and *bill*, even though they are not related to *restaurant* by hyponymy, antonymy, or other structural semantic relations (Croft and Cruse, 2004; Schank and Abelson, 1975).

Humans tend to categorize things. Lakoff (1987, pp. 113–114) suggested four types of cognitive models that are characteristics for the categories: *propositional models* that specify elements and their properties and relations, *image-schematic models* that specify schematic images, such as trajectories, long thin shapes, or containers, *metaphoric models* of mappings to other domains, and *metonymic models* that relate to using an object as an example of a group. Rosch (1978) claimed that categories do not have clear-cut boundaries but they are continuous. To be able to make distinctions between categories, *prototypes* can be used. A prototype is an average member of the category, a clear case of the membership to the category. The more prototypical a category member is, the more attributes it has in common with the other members of the cate-

gory. A prototype may even be an abstraction of characteristic features (Smith and Medin, 1981). For example, a penguin is not a prototypical bird because it cannot fly.

Not only single words but also co-occurring words have their specific meaning as a group. *Collocations* are two or more words that occur together, forming a conventionalized expression. A related term is *idiom* which means co-occurring terms that have a meaning beyond the individual meanings of the constituting terms. Collocations and idioms can be located automatically with co-occurrence analysis (Thanopoulos *et al.*, 2002), although false positives decrease the performance.

Words are something that are quite permanent in a language. A small amount of the vocabulary changes, as discussed in the beginning of the section, but most of the vocabulary is understood by every speaker of the language and will be so for a long time. In contrast, a new sentence is created almost every time someone says or writes something. Word meaning is fixed, at least at some level, but sentence meaning is something that has to be built every time one is uttered.

Pragmatics is a field of research that considers the actual language use in the real-world context. Usually an entire sentence, or several sentences, are the target of a study. The utterer of a sentence may have totally other things in mind than what the utterer actually said and the listener anyway usually understands the intention. Some examples are 'the door is open' as a request to close it, or 'the car is very dirty' as a wish to clean it. A very good textbook about the topic is by Levinson (1983). One interesting detail is the possibility of both literal and non-literal meaning of a sentence. The latter may be a metaphor, irony, or something related, which is sometimes difficult for a human to understand from a written form of language where face to face information or information about intonation are not available.

2.2.1 *Computational methods to approach semantics*

One of the main differences between computers and human beings is that computers lack the knowledge about the world. Every human being knows that a dog and a cat are special cases of an animal (Saeed, 1997, p. 68) but for a computer, it is quite difficult to know that a cat is an animal without an ontology or another resource. For a text processing system that has only text as input data, creating the actual meaning of words is almost impossible. From a computer point of view, natural language in electronic textual form is a sequence of symbols from a finite set of symbols. A computer does not understand that the similarity between 'not' and 'knot' is not as close as between 'cat' and 'cats'. Further, without any preprocessing or context, even 'cat' and 'cats' are two distinct objects.

Human beings learn the skill to use language in their childhood. First the speech consists of learned words and short sentences but gradually children start to create sentences of their own, new sentences that they have not heard from other people. At the same time children learn the physical, cultural, social, etc. facts of their surroundings. Adults are totally fluent in producing sentences

never produced before in the world, and safely sure that their utterance can be understood by other speakers of the same language. For a computer, most of this grounded information is missing if it was not specifically added in the form of rules, dictionaries, thesauri, ontologies, etc.

Computational methods can be used in solving various tasks involving semantics. A coarse-grained division of words can be done by clustering or classifying them according to their syntactic role in a sentence: part-of-speech, such as verbs, nouns and adjectives. A more detailed approach is the task of *named entity recognition* (NER) (Tjong Kim Sang and De Meulder, 2003; Lin *et al.*, 2003) to find all the objects from different classes, such as *person*, *organization* or *date*. In a related task *coreference resolution*, the goal is to identify which mentions refer to which real-world entities (Ng, 2008), for example, within a document, the phrase 'Mrs. Robinson', her whole name, and word 'she' are interpreted as the same person. Another problem to solve is to locate and disambiguate homonyms and polysemous words. In *word sense disambiguation* (WSD) (Schütze, 1992), the meaning of a polysemous word is selected from a set of alternatives based on the context of the word. A classical example is the word 'bank' which may mean monetary institution in a finance-related text or a steep slope in a text describing places in the environment. These kinds of polysemous words are problematic in many NLP applications, such as machine translation and information retrieval.

As discussed above, defining automatically the meaning of words and sentences, or even a hint of it, to enable human–machine conversation, question–answering, high-quality information retrieval and other NLP tasks, is very difficult. So far, the problem of understanding semantics by computers is far from solved. However, many steps towards understanding have been taken. Three main approaches to adding the meanings of words and sentences to a computing system can be found: manually constructed ontologies, symbol grounding, and analysis of word contexts in large corpora. These approaches are discussed in the following sections.

2.2.2 *Ontologies*

An ontology represents relationships between concepts. Ontologies may act as a knowledge base of terminology and their relations. An ontology is a directed graph that consists of concepts as nodes and relations as edges between the nodes. A well-defined ontology also has labels for the concepts and it specifies what kind of relation there is between the concepts (see, e.g., Wong, 2009). The relations may be anything sufficient for the case, such as 'is-a' or 'is-part-of' relations. To represent ontologies in computing systems, formal languages have been developed, such as DARPA agent markup language (DAML) or Web ontology language (OWL). Many domain-specific ontologies exist, such as *medical subject headings* (MeSH) (National Library of Medicine, 1960) or *Finnish geo-ontology* (Henriksson *et al.*, 2008). *WordNets* are ontologies of general language that map words and their meanings to other words using a set of relations. They are available in a range of languages, including the original English *WordNet*

(Miller, 1995; Fellbaum, 1998), Finnish (Lindén and Carlson, 2010) and other WordNets for over 60 languages[1].

During the last decade or so, ontologies have been widely used in various NLP applications as a measure of semantic relatedness (Budanitsky and Hirst, 2006). Some examples are word sense disambiguation (Yuret and Yatbaz, 2010), automatic annotation of images (Ruotsalo *et al.*, 2009), text difficulty assessment (Duran *et al.*, 2007), and monitoring disease epidemics by analysing textual reports from the Web (Steinberger *et al.*, 2008). However, in general-domain applications or within another specific domain, the strict definitions of concepts may not be applicable anymore. Ontologies cannot handle the variations and subjectivity in natural language and are thus difficult to apply to colloquial texts, such as blog texts or discussion groups. Moreover, ontologies have to be constructed mostly manually, which is laborious work, and also maintained with the evolving language and increase of knowledge, thus making them even more laborious. Lately, also automatic methods for constructing ontologies have been proposed (see reviews in: Biemann, 2005; Wong, 2009).

Other related semantic knowledge bases are taxonomies, dictionaries and thesauri. A *taxonomy* is a simple form of an ontology: it consists of concepts and their hierarchical relations. *Thesauri* contain synonyms and sometimes antonyms. *Dictionaries* give definitions of terms. All of these are manually collected knowledge bases. As another approach, Schank and Abelson (1975) prepared *scripts* to add background knowledge for a computing system. The scripts contained detailed information in simple sentences what happens, e.g., when a person goes to a restaurant.

2.2.3 *Symbol grounding*

The symbol grounding problem considers the difficulty of attaching meaning to the lexical forms of words. For a computer, symbol grounding is like trying to learn Chinese from a Chinese–Chinese dictionary, without any prior knowledge of the characters (Harnad, 1990). The question is whether this grounding can be made autonomously, without the use of semantic resources provided by humans. Solutions for the problem have been explored within the use of artificial agents: a variety of strategies aim to get the agents interpret natural language commands (Taddeo and Floridi, 2005). An example is a system that learns to follow navigational natural language directions, grounded with a map (Vogel and Jurafsky, 2010). In the robotics community, the problem is referred to the *anchoring* problem (Coradeschi and Saffiotti, 2000). Steels (2008) states that the symbol grounding problem has been solved with the use of *embodiment*: physically embodied autonomous agents that are present in the world by having a body, the possibility to move and a variety of sensors and actuators, as well as the means of signal processing and pattern recognition.

1 `http://www.globalwordnet.org/gwa/wordnet_table.html` (Accessed April 5, 2012)

2.2.4 *Vector space models using word contexts*

In contrast to ontologies and other manually constructed knowledge bases, *vector space models* (VSM) (Salton, 1971) collect the information about the meaning and the use of a word from the contexts in which it occurs. To obtain this information, large text collections are analysed with automatic methods. In a resulting *vector space*, similar items are close to each other, and the closeness can be measured using vector similarity measures. This approach has been used in Publications III, IV, V and VI and is thus discussed here in detail.

Vector space models are based on word co-occurrence calculations in a context window. The *context* may be short, the preceding and the following word, or longer, such as five preceding and five following words, a sentence, a paragraph, or the whole document. Short context such as a sentence yields syntactic features and similarities, whereas the longer contexts on document level bring on topical features (Honkela *et al.*, 1995; Sahlgren, 2006b, p. 112). Vector space models involve two main kinds of similarity: first- and second-order similarities. The *first-order similarity* is collected for a *target word* counting the frequencies of the *context words* co-occurring with the target word within a context window. For example, let us consider some example phrases containing a target word 'fruits', collected from the Europarl corpus (Koehn, 2005):

> . . . peel of oranges and other **fruits** and that we have it . . .
> . . . products such as oranges, citrus **fruits** and other produce to be . . .
> . . . our rice and our tropical **fruits** at risk. Most especially, this . . .
> . . . or producers of certain citrus **fruits**. As has been pointed out, . . .
> . . . withdrawals to 5% for citrus **fruits**, 8.5% for apples and pears . . .
> . . . very large quantities of processable **fruits** and vegetables with the result . . .
> . . . can already see the first **fruits** of this action. In connection . . .
> . . . tomatoes, peaches, pears and citrus **fruits** and also addressed the question . . .
> . . . very prosperous and enjoy the **fruits** of that prosperity. This is . . .
> . . . think we are seeing the **fruits** of that method of working . . .
> . . . and where the production of **fruits**, vegetables and wines is experiencing . . .
> . . . such a significant sector as **fruits** and vegetables, and I would . . .
> . . . be a mistake to include **fruits** and vegetables amongst all the . . .

The first-order similarities of the target word 'fruits', collected from a 5+5 context window, would be as shown in Table 1. This is one line of a word–word matrix that consists of target words ('fruits') and the frequencies of their context words. Usually the most common words (e.g., 'of', 'the', etc.) are removed or given a small weight because they co-occur practically with every target word. With the first-order similarities, collocations and compound words can be found. This similarity is also called *syntagmatic* association between words (Rapp, 2002) or *associative* similarity (Sahlgren and Karlgren, 2009). A good example of the first-order relationship is between the words 'citrus' and 'fruits'.

The *second-order similarity* finds words that co-occur with the same target words; words that are semantically related (Ruge, 1992). If there is another word

Table 1: Part of a word–word matrix.

	of	oranges	and	that	citrus	vegetables	the	⋯
⋯	⋯							
fruits	8	2	14	3	4	4	7	⋯
⋯	⋯							

'trees' that co-occurs with both 'citrus' and 'oranges', we can say that 'fruits' and 'trees' have the same context and form a second-order relationship. In other words, 'trees' could be a replacement for 'fruits' in the Europarl examples. The second-order similarity can correspond to any kind of semantic similarities between words. They are also called *paradigmatic* associations (Rapp, 2002). The second-order similarities can be observed either by collecting a word–word matrix, where the values are co-occurrences of words within some contexts, or a document–document matrix, where the values define how many common features the documents have. Schütze and Pedersen (1997) were the first ones to use the co-occurrence-based second-order statistics in vector space models.

These first- and second-order similarities can be used to build a vector space which has semantically similar words close to each other. VSM relies on word distributions instead of human knowledge and can be constructed quickly without manual annotation of data or any restrictions with the domain or language. The vector space model has been used in many applications, such as information retrieval (Salton *et al.*, 1975), text categorization (Lewis, 1992), word sense disambiguation (Schütze, 1992), cross-document coreferencing (Bagga and Baldwin, 1998), and bilingual lexicon acquisition (Sahlgren and Karlgren, 2005; Rapp and Zock, 2010). Bullinaria and Levy (2007, 2012) analysed the effect of different context windows, distance measures, preprocessing and dimensionality reduction.

2.3 MULTILINGUALITY

Traditionally, most of the NLP methods have been developed for the English language. Only quite recently, other languages have gained more interest, but still English is overwhelmingly most popular in two of the main conferences on computational linguistics (Bender, 2011). The Internet has the largest collection of multilingual documents. W3Techs counts the languages of Web sites in the world[2]. According to their measures in April 2012, English was the most popular with 56% of Web sites, and the following languages were German (6.6%), Japanese (4.9%), Russian (4.8%), Spanish (4.6%), Chinese (4.4%) and French (4.1%). Also among the Internet users in 2011 by L1 (native) language, English was the most popular with 27% of users, followed by Chinese (24%), Spanish (8%), Japanese (5%), Portuguese (4%), German (4%) and Arabic (3%)[3].

2 W3Techs `http://w3techs.com/technologies/overview/content_language/all` (Accessed April 5, 2012)

3 Internet World Stats, `http://www.internetworldstats.com/stats7.htm` (Accessed April 5, 2012)

In order to process automatically other languages than English, a great deal of differences have to be taken into account. Besides the obvious differences between the vocabularies, also, for example, the number of words required to say the same utterance in different languages vary a lot. There are also large differences between unique word tokens occurring in a large text corpus of texts translated to different languages. The larger the number of unique tokens, the more complicated are the automatic processes to identify the words. The differences can be seen for example in the Europarl parallel corpus (version 3) of European parliament plenary speeches translated (manually) into 11 languages. In a sentence-aligned test set (Koehn *et al.*, 2003) having the same information in each language, the number of words and unique word tokens are shown in Table 2. The number of words varies between 203 000 words in the Finnish translation to 326 000 words in French. The number of unique word tokens, i.e., the number of different words in the corpora, vary from 12 600 unique word tokens in English to 37 000 tokens in Finnish.

Table 2: The Europarl corpus languages, language groups (Romance (R) and Germanic (G)), and number of words and number of word types in a test set after preprocessing.

Language	**Abbr.**	**Group**	**Words**	**Unique**
French	fr	R	326k	16.4k
Greek	el		322k	23.0k
Spanish	es	R	309k	18.4k
Portuguese	pt	R	303k	18.1k
English	en	G	299k	12.6k
Dutch	nl	G	299k	17.6k
Italian	it	R	291k	18.3k
German	de	G	274k	22.9k
Danish	da	G	272k	20.4k
Swedish	sv	G	268k	21.9k
Finnish	fi		203k	37.0k

One reason for the small number of unique words in English is the small amount of affixes (e.g., 'work', 'work+s', 'work+ed', 'work+ing') and the interpretation of the same words in different parts-of-speech: 'to work' (*verb*) and 'the work' (*noun*). Finnish as the other extreme is an agglutinative language in which the same base word may get hundreds of different surface forms (e.g., Helsinki, Helsingissä, Helsinkini, Helsinkikään, etc.).

One more reason to the small number of unique words in English are the word compounds. They do exist in English but are very rare compared to some other languages, such as German or Finnish. The simplicity of processing written English compared to many other languages may be one of the reasons why NLP research stayed within the English language for a long time, while another reason is the largest amount of evaluation data existing for English.

Multilingual data evaluation conferences, such as TREC[4], CLEF[5] and NTCIR[6], have started to actively collect evaluation corpora suitable for multilingual and language-independent approaches. Also SemEval workshops (Agirre *et al.*, 2007; Erk and Strapparava, 2010) have had many multilingual tracks with evaluation data. *Parallel corpus* is a text collection that contains the same information in at least two languages, having the texts aligned at paragraph or sentence level. In the former, the alignment is usually 1–1 and in the latter, it may be 0–n. Examples are the Europarl corpus (21 languages in version 6)[7] (Koehn, 2005), Universal Declaration of Human Rights[8] (more than 300 languages and dialects) (Vatanen *et al.*, 2010), and the Bible (426 languages) (Chew *et al.*, 2006). Parallel corpora make it possible to compare the performance of NLP systems between languages. *Comparable corpora*, such as Wikipedia, contain texts about the same topics, but they are not direct translations of each other. *Multilingual corpora* is any of the previous, or any other corpus that contains texts in more than one language. *Bilingual corpus* contains texts in two languages, usually parallel documents.

2.3.1 *Language families*

It is difficult to say how many languages there are in the world, because the boundary between separate languages or separate dialects is not clear. One basic rule is whether two speakers understand each other when speaking their own languages. If yes, they speak the same language and the possible differences may arise from different dialects. However, the state borders have a role in the distinction: for example, Danish, Norwegian and Swedish are mutually intelligible (Delsing and Åkesson, 2005) but they are spoken in different countries and treated as different languages. The estimated number of languages in the world is in the thousands (Katzner, 2002), perhaps about 7000 (Gordon and Grimes, 2005). However, only about 400 languages have more than 1 million speakers (Gordon and Grimes, 2005).

Languages of the same origin can be grouped into language families. Language families and groups share many words and syntactic features, and have been shown to perform similarly in NLP applications. For example, machine translation is easier within Germanic or within Romance languages than mixed or involving Greek or Finnish (Koehn, 2005), and Germanic language pairs Danish–Swedish and English–Swedish correlate better than for example Finnish as one of the languages (Publication VI). In cross-lingual information retrieval, results within the Indo-European family were better than those involving Arabic (Chew and Abdelali, 2007) which belongs to the Afro-Asiatic family.

In order to give some background how the languages mentioned in this dissertation relate to each other, some of the major languages in the largest language families and groups are listed in Tables 3 and 4, following Katzner (2002,

4 http://trec.nist.gov/
5 http://www.clef-campaign.org/
6 http://research.nii.ac.jp/ntcir/index-en.html
7 http://www.statmt.org/europarl/
8 http://www.un.org/en/documents/udhr/

pp. 2–9). While language families are a linguistic categorization (with divergent opinions), similar categorizations can be found using only statistical methods to analyze the written form of the languages: Sadeniemi *et al.* (2008) analysed 21 European languages in the Germanic, Romance, Slavic, Hellenic, Celtic, Baltic, Finno-Ugric and Semitic language groups with statistical methods. They managed to group languages according to their linguistic groups by analysing the morphology and word order. Also individual languages can be identified with automatic methods using only a short example, 5–21 characters of the unknown language (Vatanen *et al.*, 2010). The approach did not make assumptions about the languages, such as word boundaries, and was tested for 281 languages.

2.3.2 *Writing systems*

Only few years back, separate character encodings were used for each language in automatic processing of text. Today Unicode has, at least partly, solved the problem by providing a common base for most of the scripts and thus most multilingual document sets can be handled with only one coding system. However, knowing some characteristics of the writing systems in different languages is essential when building a multilingual text processor.

The Latin alphabet is widely adopted in many languages. It has quite accurate correspondence between the characters and the phonemes (consonants and vowels) – English and French are quite obvious non-prototypical languages in this group. Many European languages use their own additional letters, such as 'ä' and 'ö' in German, Swedish and Finnish, and 'ł' in Polish. Other popular scripts in Europe are the Greek and Cyrillic alphabets.

Many African languages are written with the Latin alphabet, but in Asia there is a rich diversity of different scripts. The Arabic system consists of basically consonant characters, whereas vowels are marked as vowel signs above or below the letters. However, the use of diacritic dots and vowel signs vary a lot (Versteegh, 1997). The Arabic script is used also in many other languages, e.g., Persian and Urdu.

Many Asian scripts are syllable-based, such as Japanese and Chinese ideographs. In the Chinese language, each word consists of one or more syllables, each syllable marked with one Chinese character. The word boundaries are not marked in any way and thus many NLP systems for Chinese start with identifying the word boundaries. Written Japanese combines Chinese characters and two own syllabic scripts. Written Korean characters consists of syllables of single or double consonants or vowels.

In addition to the various uses of characters, also the writing order of the characters vary. From left to right is common within texts written in the Latin alphabet, but for example Arabic, Persian and Hebrew are written from right to left. Some languages, such as Chinese, Japanese and traditional Mongolian, can be written from top to bottom. Also more complicated writing orders exist (Katzner, 2002).

Table 3: Language families according to Katzner (2002, pp. 2–9).

Family	**Subgroup**	**Major languages**
Indo-European	Germanic	*Western:* English, German, Yiddish, Dutch, Flemish, Afrikaans; *Northern:* Swedish, Danish, Norwegian, Icelandic
	Italic	Latin
	Romance	Italian, French, Spanish, Portuguese, Romanian
	Celtic	*Brythonic:* Welsh, Breton; *Goidelic:* Irish, Scottish
	Hellenic	Greek
	–	Albanian
	Slavic	*Eastern:* Russian, Ukrainian, Belorussian; *Western:* Polish, Czech, Slovak; *Southern:* Bulgarian, Serbo-Croatian, Slovenian, Macedonian
	Baltic	Lithuanian, Latvian
	–	Armenian
	Indo-Iranian	*Iranian*: Persian, Pashto, Kurdish, Baluchi, Tajik, Ossetian; *Indic*: Sanskrit, Hindi, Urdu, Bengali, Punjabi, Marathi, Gujarati, Oriya, Bhojpuri, Maithili, Magahi, Rajasthani, Assamese, Kashmiri, Nepali, Sindhi, Sinhalese
Uralic	Finno-Ugric	*Finnic:* Finnish, Estonian, Mordvin, Udmurt, Mari, Komi; *Ugric:* Hungarian
	Samoyed	
Altaic	Turkic	*Southwestern:* Turkish, Azerbaijani, Turkmen; *Northwestern:* Kazakh, Kyrgyz, Tatar, Bashkir; *Southeastern:* Uzbek, Uigur; Chuvash
	Mongolian	Mongolian
	Tungusic	
Caucasian	Southern	Georgian
independent		Basque
Dravidian		Telugu, Tamil, Kannada, Malayalam
Munda		
Sino-Tibetan	Sinitic	Chinese
	Tibeto-Burman	Burmese, Tibetan
Tai		Thai, Lao
Miao-Yao		
independent		Japanese
independent		Korean
Mon-Khmer		Vietnamese, Khmer

Table 4: Language families according to Katzner (2002, pp. 2–9).

Family	**Subgroup**	**Major languages**
Austronesian	Western	Indonesian, Malay, Javanese, Sundanese, Madurese, Tagalog, Visayan, Malagasy
	Micronesian	
	Oceanic	Fijian
	Polynesian	Maori, Tongan, Samoan, Tahitian
Papuan		
Australian		
Paleo-Asiatic		
Eskimo-Aleut		Eskimo
Niger-Congo	Mande	
	Atlantic	
	Gur	
	Kwa	
	Kru	
	Adamawa-Ubangi	Adamawa, Ubangi
	Benue-Congo	Nigerian, Bantu, Swahili
Afro-Asiatic	Semitic	*North Arabic:* Arabic, Maltese; *Canaanitic:* Hebrew; *Aramaic:* Syriac, Aramaic, Assyrian; *Ehtiopic:* Amharic, Tigrinya, Tigre, Gurage, Harari, Ge'ez
	Berber	Kabyle, Tachelhit, Tamazight, Riff, Tuareg
	Cushitic	Somali, Oromo, Sidamo, Hadiyya, Beja, Afar
	Chadic	Hausa
	Omotic	Wolaytta
	Egyptian	Coptic
Chari-Nile	Eastern Sudanic	Nubian; *Nilotic:* Luo, Dinka, Nuer, Shilluk, Lango, Acholi, Alur, Teso, Karamojong, Masai, Turkana, Bari, Lotuko, Kalenjin, Suk
	Central Sudanic	Sara, Mangbetu, Lugbara, Madi
Saharan		Kanuri, Teda
Maban		Maba
Khoisan		Hottentot, Bushman, Sandawe, Hatsa
North American Indian		e.g., Cree, Navajo, Apache, Sioux, Crow Cherokee, Yuma, Choctaw, Zapotec, Maya
Central and South American Indian		e.g., Guaymi, Cuna, Carib, Guarani, Jivaro, Goajiro, Araucanian

2.3.3 *Syntax or grammar*

Syntax of a language defines how words can be put together to form a sentence. Within the languages in the world, all categorize words at least into nouns (N) and verbs (V) (Hockett, 1963). Other important *parts-of-speech* are adjectives, pronouns and numerals. A simple sentence consists of a subject (S), a verb (V) and possibly an object (O), denoting roughly the subject which is doing something, the action done, and the object of the action. Subject is often a noun or a noun phrase (NP) which contains a noun or a noun-like word and optionally determiners (such as articles or numerals) and modifiers (such as adjectives). NP chunks are widely used in many NLP applications, for example as candidates of keyphrases. The object of a sentence may be one word, such as adjective or pronoun, or a longer phrase, such as noun phrase or a content clause. Among the languages in the world, many restrict the order of subject, verb and object in a sentence. SOV, SVO, VSO, and VOS orderings are all common but OSV and OVS are very rare (Pullum, 1977). SOV, the most typical order, is used in, e.g., Basque, Hindi and Japanese. SVO is the second most typical order and used in, e.g., English, Finnish, Russian and Spanish. VSO is commonly used in Arabic and Celtic languages. However, many languages, including Finnish, are not strict in this rule and allow several orderings.

In many languages, the words that constitute a sentence, can be split into smaller units. *Morphology* studies the smallest meaningful linguistic units of language: word stems and possible affixes: prefixes, infixes and suffixes, e.g., 're'(*prefix*) + 'play'(*stem*) + 'ed'(*suffix*). There are two kinds of morphology: inflectional and derivational. *Inflectional* morphology means adding prefixes, infixes or suffixes to a word when using it in a sentence, such as 'cat' → 'cats'. Derivational morphology is obtaining a new word by adding an affix to a word, such as 'write' → 'writer'. Some languages, such as Chinese, do not have morphology at all. Examples of morphologically rich languages are Arabic, Finnish and Turkish. A good introduction into morphology can be found in Creutz (2006).

2.3.4 *Differences in semantics between languages*

When considering the languages around the world, starting from languages spoken in rainforests, going through Latin, and visiting the modern high-tech English, it is evident that the use of the various languages is nowadays, and has been earlier, very different from each other. The chief interests of people have a substantial role in the words developed to a language and in categorization of the words (Boas, 1966). One example is the colour naming in different languages studied by Berlin and Kay (1969): even though people can distinguish between different colours all around the world, some languages have much less basic colour names than others. All languages have at least two basic colour names, 'black' and 'white'. In some languages these are the only two basic colour names, and they divide the colour space into two parts: 'dark' (for black, blue, green and grey) and 'light' (for white, yellow, orange and red). If there is also a third colour in a language, it is always 'red'. In the case of four

basic colour names, the fourth name is either 'yellow' or 'green'. The fifth name is the remaining of the two names. The sixth name is 'blue' and the seventh 'brown'. Only after them come 'purple', 'pink', 'orange' and 'grey'.

Many words related to a local culture, such as Finnish 'mämmi' (the Finnish Easter pudding of great taste and looks) or 'sampo' (a magical object in the Finnish mythology that brings good fortune for its owner) are part of everyday language for those living within the cultural environment but may not be known even at the conceptual level in other languages. Saeed (1997) refers to many works studying differences across languages related to, e.g., meanings of grammatical roles. English verbs for putting on clothes do not make difference on the body parts involved, as do, for example, Japanese and Korean (Saeed, 1997, pp. 41–43). There are also differences across languages on how wide or detailed certain concepts are. An example is the word 'wood' in English, having the meanings of a small forest and the substance found in trees. In German there are two distinct words for these, 'Wald' and 'Holz', respectively. However, the word 'Wald' has also the meaning of a large forest, which cannot be referred to as a 'wood' in English, as discussed by Alansary *et al.* (2006). The differences between semantic interpretation in languages have a substantial role in translating between languages. Also NLP systems constructed in a multilingual setting face the same problem.

2.3.5 *Language universals*

Language universals are part of *linguistic typology* research. While not participating here in the discussion about the role of language universals in revealing something from human cognition and mind (Greenberg, 1966), it is to be noted that language universals contain very interesting features that are general within all languages and which could potentially be used in a multilingual setting of NLP. So far language universals have not gained much interest in the field of statistical NLP – an interesting exception is the work by Schone and Jurafsky (2001).

Although languages have many differences – which is why a new language is difficult to learn – there are many features that are shared among (more or less) all languages in the world. Every language has a vocabulary that contains words for the most important things to a human being. For example, among the studied languages in the world, all languages have terms for *eyes*, *nose*, *mouth*, *toe* and *finger* (Brown, 1976; Andersen, 1978). If there is a separate term for *foot*, then there is also one for *hand* but not vice versa (Brown, 1976). All languages distinguish between male and female parent (*father* and *mother*) by separate terms (Greenberg, 1966, Sec. 5), have first and second person singular pronouns and have proper nouns (Hockett, 1963). All languages have words comparable, though not necessarily identical in meaning, with *cry* / *weep* and *smile* / *laugh* (Wierzbicka, 1999). Conventionalized metaphors tend to be similar across languages (Croft and Cruse, 2004, p. 195), which is due to their cognitive significance which is grounded in human experience (Lakoff, 1993, p. 229). This kind of information might appear very useful in many multilingual NLP tasks in which statistical features, such as word frequencies and

co-occurrences, could be combined with the information of what should be present in the language. Applications could be, for example, in automatic translation between languages for which there are no resources such as translation examples available, or in tasks related to automatic semantic analysis of language.

Language universals have been collected to the *Universals archive* (Plank and Filimonova, 2000). The archive[9] contains over 2 000 entries that are intended to apply to all known languages in the world. Besides lexicon-related universals mentioned above, the universals archive contains also universals about syntax, morphology, phonology, phonetics, and semantics. It is to be noted, however, that not all the universals apply to every single language in the world.

9 http://typo.uni-konstanz.de/archive/

3

COMPUTATIONAL METHODS FOR LANGUAGE INDEPENDENCE

In the previous chapter, various characteristics of natural languages as data and computational approaches to deal with them have been discussed. In this chapter, the concept of *language independence* in NLP is introduced and the variations in it are discussed. The chapter continues with presenting machine learning approaches and discussing those methods and approaches which can be applied in a language-independent setting. Instead of providing a comprehensive list of the wide range of machine learning methods, only methods having importance for this work and other language-independent approaches in the literature are discussed.

3.1 LANGUAGE INDEPENDENCE IN NATURAL LANGUAGE PROCESSING

Natural language processing systems are used and developed for multiple languages. The need of language-independent systems arises from multilingual environments in which it would be too expensive to build an individual system for each language separately. Such environments are, for example, the Web, and those societies, companies and governments that use multiple languages. Another area is non-standard usage of languages, such as colloquial language in textual form (e.g., in blogs, IRC, forums), dialects, and ancient versions of languages. There is also a multitude of languages with few speakers for which language-specific resources are not available. Finally, research on language-independent methods may reveal something new about *language universals* (Greenberg, 1966) or human language in general.

A method is fully language-independent if it can be applied to a corpus in a new language without any modifications of the system or system parameters. Some assumptions have to be made: the language needs a conventionalized written form in which there is enough data available, and the language should be writable with an automatic text processing system, using e.g., Unicode characters. Also the existence of a concept *word* that applies cross-linguistically is required (Bender, 2009). Some kind of preprocessing for the text has to be accepted, in order to collect the text content from noisy documents, such as Web documents with advertisement and pictures. Also, punctuation requires specific processing separate from words.

Even language-independent methods need some indication about the characteristics of the language. A simple case is a changed value in a parameter, or a corpus about the general usage of the language which is rather easy to obtain. In methods with lower language independence, the additional language-specific information is added to the original data in the preprocessing phase (as parts-of-speech, for example), or directly as part of the learning system. The

choice of language independence mostly relies on the preprocessing of data, the use of additional information, and the applied machine learning paradigm.

Statistical unsupervised machine learning methods for textual data, often referred to as text mining, are language-independent by default. While supervised learning methods require labelled data, unsupervised methods rely on statistical dependencies in the data. Different machine learning paradigms are discussed in Section 3.2.

3.1.1 *Levels of language independence*

Language independence is a continuum from total independence of the used language to a fully language-specific system. In the following, the levels of language independence are introduced. Example methods and applications, collected from this dissertation and from the literature, are categorized according to their level of language independence.

FULL INDEPENDENCE OF LANGUAGE Methods that are fully independent of the used language do not require language-specific resources and can be applied to text in a new language as-is. Examples of fully language-independent methods do detection of sentence boundaries (Kiss and Strunk, 2006), syntactic cluster induction (Clark, 2000) and labelling syntactic clusters (Schone and Jurafsky, 2001). One possibility to ground language-independent methods is the use of other non-linguistic information sources, such as images, Web links (Brin and Page, 1998), or videos. The approach is also referred to as *multimodality*, but it has been studied mainly from the viewpoint of augmenting images or videos with text (Feng and Lapata, 2008; Koskela *et al.*, 2009), rather than grounding text with visual elements.

MONOLINGUAL CORPUS To know how a language is used in general, an additional monolingual corpus can be applied. The use of a monolingual reference corpus is highly language-independent if the corpus does not require any language-specific preprocessing. The justification of the language independence of this approach is that if NLP tools are needed for a certain language, it is assumed to be large enough to have also some other text collections easily available. Example approaches are spellchecking and autocorrection (Whitelaw *et al.*, 2009) and keyphrase extraction (Publication I).

TUNING A PARAMETER Tuning one or more parameters of a system for increasing the performance for each language is a small step towards language specificity, which also needs manual efforts. However, languages within a single language group or family may possibly accept similar parameter values (Bender, 2009) and thus ease the manual work. Example of tuning a parameter are selecting a threshold value in morpheme segmentation (Creutz and Lagus, 2007), keyphrase extraction post-processing (Publications II, III), selecting the number of keyphrases (Publication III) and selecting the number of clusters (Publication III).

BOOTSTRAPPING Bootstrapping from a small number of seeds requires few examples in each language. However, these examples have to be labelled or at least selected manually for each language separately. These approaches lie between language-independent and language-specific methods. Examples vary from named entity recognition (Cucerzan and Yarowsky, 1999), parsing (Haghighi and Klein, 2006) to topic detection (Zagibalov and Carroll, 2008).

RESOURCES FOR HUMAN USE The use of resources prepared originally for human use are also on the borderline between language independence and language specificity. If the resources have been prepared for humans in one language, it is possible that similar resources have been produced for other languages as well. The approaches applying these resources vary from statistical methods using a bilingual corpus in vector space evaluation (Publication VI) to many rule-based approaches that need a heavy manually constructed system how the resource, such as dictionaries, thesauri or controlled vocabularies are used in each application separately. It is easy to say that the language independence is higher in the former and much lower in the latter, due to the cost of rebuilding the rules for a new language.

LANGUAGE SPECIFICITY Language-specific methods require resources prepared originally for computer use, such as stop word lists or part-of-speech taggers. The resources may or may not be helpful for humans as well. Examples of human-readable resources are manually labelled data and ontologies such as *WordNet* (Miller, 1995; Fellbaum, 1998). Examples of language-specific methods are, e.g., rule-based stemmers, such as *Porter's stemmer*[1], the other stemmers in the Snowball package[2], many lemmatizing and part-of-speech tagging systems and systems that use manually annotated corpora, dictionaries and thesauri. The systems may also make strong assumptions about the language at hand, for example being non-inflectional or having a certain word order. The development of these systems needs substantial manual expertise and effort for each language separately.

3.1.2 *Discussion*

A great part of the NLP research still today concentrates on the English language and the problems related to automatic processing of language in general have not been solved yet. In the literature, there are many well-performing NLP methods for English but their performance with other languages is usually much poorer because English happens to be an easy language for many tasks. It is quite difficult to port methods developed for English to new languages. For example, Nivre *et al.* (2007) note that many studies have reported a substantial increase in error rate when applying a statistical parser developed for English to other languages (Czech, Chinese, German, Italian). In contrast, language-independent methods may work for many languages but apparently some language-specific information would increase the performance (Kim *et al.*, 2010b).

1 http://www.tartarus.org/~martin/PorterStemmer
2 http://snowball.tartarus.org/

Some studies claim to be language-independent (e.g., Gómez-Soriano *et al.*, 2005) but the claim is not valid if the tested languages come from a single language family (the language families were discussed earlier in Section 2.3.1). It is possible that the method uses or has found features that happen to apply to a certain type of languages only (Bender, 2009). Furthermore, language independence has been claimed in a multilingual setting, in which a task is performed for several languages but various language-specific tools are used for each language separately (Tjong Kim Sang, 2002; Nivre *et al.*, 2007). These approaches could be called cross-lingual or multilingual to emphasize the fact that the systems need a lot of language-specific processing.

As discussed in Section 2.3, there are notable differences between languages. Bender (2009) claims that a truly language-independent system should work equally well across languages. However, as people find some tasks more difficult in one language than in another (for example, English letter-to-phoneme conversion compared to, say, German), why should a machine be able to perform equally well in all of them?

3.1.3 *Domain independence*

The domain of text documents vary within the large range of specific and expert domains. The effect of a domain to the vocabulary and word distributions of a text was discussed in Section 2.1.2. It is quite common to build a domain-specific application instead of a general-domain one because the vocabulary is more restricted and stylistic variations are smaller. However, domain specificity prevents from applying the method outside the domain in which it was developed.

A *domain-independent* method can be applied to a new domain without using new resources and without a substantial decrease in performance. Practically this means that no additional domain-specific resources, such as domain-related corpora, thesauri, word lists or annotations, are required for training the method. If a method is language-independent, it usually does not use any domain-specific information.

Within the methods presented in this dissertation, all approaches are domain-independent. The *Likey* keyphrase extraction method is applied to encyclopedia texts on several topics in different domains (Publications I, III), scientific articles in computer science (Publication II), and professional and lay texts in the medical domain (Publication VII). The document clustering approach is tested with four datasets of various domains and genres, including newspaper texts and newsgroup postings about business, health, politics and cars (Publication V). Also the other approaches are domain-independent but tested on one domain only.

3.2 MACHINE LEARNING AND STATISTICAL METHODS

Machine learning methods have found their place in the NLP community. Unlike many traditional NLP approaches, *machine learning* (ML) methods typi-

cally use large data sets and apply machine learning algorithms to them. ML methods construct a model to solve a problem, analyse thousands of data items and fit model parameters to correspond to the data. A simplifying example can be used to explain the procedure: The *problem* to be solved could be finding the relation between the diameter d and circumference C of a circle. In this example, different models were tried and, for instance, using an educated guess, a *model of form* $C = a \cdot d$ was selected. The *data* were drawings of circles of different sizes and the measurements of the diameters and circumferences. The single *model parameter*, constant a, was fitted to correspond the data and as the resulting *model* for the problem, the relation between the diameter and circumference of a circle $C = a \cdot d$, the *constant* a was set to $a = 3.14159$. Surprisingly or not, the constant happens to be equal to the widely known constant π.

We would not be doing research if the results were not evaluated with a set of evaluation data. In the example, *evaluation data* could be obtained by drawing another set of circles, with another pencil perhaps, and checking whether the constant $a = \pi$ applies also to them. Typical to machine learning is that the models are much more complex than in the example. Also the used data sets are usually very large, for example thousands or millions of data items. These large data sets introduce noise in modelling and thus the results are very rarely as easy to get as in this example.

Statistical methods use counts, frequencies, distributions and probabilities in data rather than applying simple rules like IF-THEN used in rule-based methods. All the events occurring in natural data are assumed to be following distributions and have some probability of occurrence. As an example, classification into separate, predefined classes does not follow the statistical view of the world, in which strict class boundaries without exceptions are very rare. However, in practical applications, simplifying assumptions about, for example, the number of clusters in the data, have to be made to ensure the system performance in the way expected.

3.2.1 *Machine learning paradigms*

The methods for machine learning can be roughly divided into supervised and unsupervised methods. The basic difference between them is that supervised methods use labelled training data to build the model, whereas unsupervised methods do not use the labels. In addition to these, various approaches that contain features from both supervised and unsupervised methods exist, such as semi-supervised and weakly supervised learning. The choice of the ML paradigm has a significant role in the language independence of a system. In this section, the machine learning methods are considered from the text processing point of view.

SUPERVISED LEARNING Supervised learning approaches fit the parameters of a model to a labelled data set. The labels show what structure should be found in the data and have usually been added manually to the data. Supervised learning involves *classification* into a discrete predefined set of classes

or *regression* in the case of continuous-valued output (Bishop, 2006). Supervised methods are first trained with a training data set and the methods can be then used for labelling new unlabelled data. One of the most commonly used supervised methods is *support vector machine* (SVM) (Vapnik, 1995) that has been successfully applied, besides other data, also to many NLP tasks, such as text categorization (Joachims, 1998), grammatical chunking (Kudo and Matsumoto, 2001) and parsing semantic role (Pradhan *et al.*, 2005). Another popular classifier is the *naive Bayes* classifier (see, e.g., Bishop, 2006). It is a simple model to use for large data sets and dimensionalities. *k nearest neighbours* (kNN) (Cover and Hart, 1967) is a non-parametric classification method: each unlabelled data item is labelled according to the labels of the majority of the nearest neighbours. Neural networks learn a (usually) non-linear mapping between input features to output labels. A commonly used version is *feed-forward artificial neural network* (ANN) (see, e.g., Haykin, 1994).

Supervised methods are very good in various classification tasks: the target classification of the training data is known, and if the training data set is large enough, of sufficient quality, and actually contains some information that can be used in the classification, the method can learn a fairly accurate mapping between data items and the classes. However, due to the need of (manually) labelled data, it is difficult to use supervised learning methods in data-driven and language-independent way.

UNSUPERVISED LEARNING Unsupervised learning methods measure the densities in a data set to perform, for example, clustering into groups emergent in the data. No labelled data or predefined sets of clusters are exploited. *Data mining* and *text mining*, data mining of textual data, belong to this learning paradigm. The strengths of unsupervised learning methods are in explorative analysis of previously unseen data and in visualizing the structure of complex data. Generally, supervised methods reach better accuracy than unsupervised methods due to the supervision on what one wishes to find, but in some tasks unsupervised methods perform as well as supervised methods or even better (e.g., Kurimo *et al.*, 2010b; Yarowsky, 1995), because of their wide coverage and the ability to generalise to new data. Fully automatic methods usually rule out supervised learning (due to the requirement of labels or tags) but can often be found within unsupervised learning approaches. Methods for unsupervised learning are discussed from the language-independent NLP point of view in Sections 3.4 and 3.5.

SEMI-SUPERVISED APPROACHES Semi-supervised learning methods are a group of approaches which combine the strengths of supervised and unsupervised learning: the accurate results from supervised learning and the ability to generalize from unsupervised learning. Semi-supervised learning uses a small set of labelled data and usually a much larger set of unlabelled data.

Self-supervised learning is a synonym for semi-supervised learning (Wu *et al.*, 2001), or may mean that the label information has been collected for a part of the instances using heuristic rules (Wu and Weld, 2010; Kim *et al.*, 2011). Related approaches, *co-training* (Blum and Mitchell, 1998) and *self-training* are

bootstrapping methods that first use supervised learning for labelled data and then classify unlabelled data with the most confident predictions, possibly in several iterations. In co-training, there are two or more classifiers where as in self-training only one classifier (Mihalcea, 2004) or a single view (Ng and Cardie, 2003) is used. These approaches are also referred to as *weakly supervised* algorithms (Ng and Cardie, 2003). In *prototype-driven* learning, a few prototypical examples are specified for each target label. This approach was used for part-of-speech tagging in Haghighi and Klein (2006). In *reinforcement learning*, a teacher gives positive or negative reward depending on whether the system did good and poor actions. The actual right answer is not given (Sutton and Barto, 1998).

3.2.2 *Probability theory*

Probability distributions

Many statistical natural language processing methods are based on word frequencies and their distributions in a document or a set of documents. Natural languages have an interesting property which is actually also common in other natural systems (Newman, 2005): the distribution of word frequencies follows the exponential distribution, often referred to as Zipf's law (Zipf, 1949), which is shown on a log–log scale in Figure 1. In the figure, word frequencies are compared to their rankings in the frequency order, calculated from sentence-aligned translations in seven European languages, a part of a test set from the Europarl corpus (Koehn *et al.*, 2003). The distributions of all the languages are almost linear on the log–log scale. This shows that the few most common words in each language are very frequent, but in the other end of the curve, there is a very long list of very rare words. The agglutinative language Finnish differs from the other languages by showing substantially smaller frequencies for separate words than the other languages.

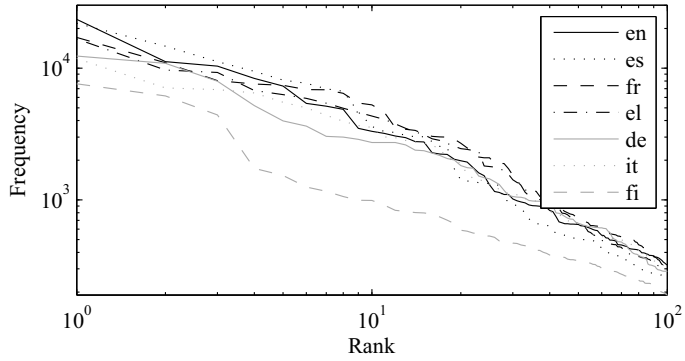


Figure 1: Word frequencies and their rankings in the frequency order on a log–log scale in English (en), Spanish (es), French (fr), Greek (el), German (de), Italian (it) and Finnish (fi).

A textbook definition of probability distributions can be found, e.g., in Bishop (2006). A popular distribution in statistical modelling is the continuous *normal (Gaussian) distribution*. The probability density function of the normal distribution for a single variable $x \in \mathbb{R}$ is defined as

$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \tag{1}$$

with mean μ and variance σ^2 as the parameters. The Gaussian distribution does not usually hold for textual data, because most of words are very rare (Manning and Schütze, 1999, p. 54).

The Gaussian distribution approximates the discrete *binomial distribution* for a large set of samples. Assume a random variable x which can attain one of two values, say, $x = 0$ or $x = 1$. Then the probability of m observations of $x = 1$ in a set of M samples is

$$\text{Bin}(m|M,\mu) = \binom{M}{m} \mu^m (1-\mu)^{M-m}, \tag{2}$$

where the probability of observing $x = 1$ is $\mu \in [0,1]$. Binomial is seen as a better approximation of text than Gaussian distribution because the process of counting words can be seen as a form of binary trials (Dunning, 1993). *Multinomial distribution* is a multivariate generalization of the binomial distribution for the observed counts m_i of k possible outcomes given a total number of observations M

$$\text{Mult}(m_1, m_2, \ldots, m_k|\boldsymbol{\mu}, M) = \binom{M}{m_1 m_2 \ldots m_M} \prod_{i=1}^{M} \mu_i^{m_i}, \tag{3}$$

where $\boldsymbol{\mu} = (\mu_1, \cdots, \mu_k)$.

Bayesian statistics

Bayesian statistics considers probabilities to be a measure of uncertainty. The prior probability of an event is updated with new data in order to obtain a more certain posterior probability of the event. The probabilities of events X and Y are given as $P(X)$ and $P(Y)$, respectively. The intersection $P(X \cap Y)$ is the probability that both X and Y occur. The conditional probability (the posterior) of an event X if we know the result of event Y is

$$P(X|Y) = \frac{P(X \cap Y)}{P(Y)}. \tag{4}$$

Bayes' theorem connects events X and Y in the following way:

$$P(X|Y) = \frac{P(X \cap Y)}{P(Y)} = \frac{P(Y|X)P(X)}{P(Y)}, \tag{5}$$

where $P(X)$ is the prior of X. The Bayesian statistics is one of the basic approaches in NLP and has been applied to many usages. Some examples are Bayesian networks in mining multilingual named entities (Nabende, 2011) and the use of non-parametric Bayesian methods in phrase alignment and extraction (Neubig *et al.*, 2011).

3.2.3 *Information theoretical measures*

Information theory (Shannon, 1948) has been the basis for many data mining and NLP approaches. The theory is founded on the idea of maximizing the amount of information transferred through an imperfect communication channel. A description of information theoretic approaches for text data can be found, e.g., in Manning and Schütze (1999). *Entropy* is an information theoretic measure of confusion. For a random variable X over a discrete set of symbols $\mathcal{X}$, entropy is defined as

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x), \tag{6}$$

where $p(x)$ is the probability of x. Entropy can be used in NLP for example in term weighting (Section 3.3.5). *Joint entropy* for a pair of discrete random variables X and Y is defined as

$$H(X,Y) = -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x,y) \log p(x,y), \tag{7}$$

where $\mathcal{X}$ and $\mathcal{Y}$ are discrete sets of symbols. *Mutual information* (MI) measures the dependence of two random variables X and Y as

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ &= H(X) + H(Y) - H(X,Y) \\ &= -\sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)\,p(y)}. \end{aligned} \tag{8}$$

MI has been used in various NLP applications, including evaluation of document clustering (Publication V), keyphrase extraction (Pantel and Lin, 2001), and as a similarity measure of two canonical variates (Publication VI).

3.3 LANGUAGE-INDEPENDENT PREPROCESSING

There are various scripts used for writing different languages, as has been discussed in Section 2.3.2. In this section, some issues of preprocessing multilingual text are considered.

3.3.1 *The dirty work: cleaning the text*

Before anything else can be done, the text to be analyzed has to be extracted from its original context: from a Web page, pdf document, e-mail, and so on. Besides the actual text content, there is usually much additional information on the electronic files, such as html tags, advertisement, page numbers and repetitive text passages such as copyright texts. These usually have to be removed in one way or another. Unfortunately, everything may not be possible to remove in a way that applies to all possible languages.

3.3.2 *Punctuation*

Punctuation conventions vary a lot among languages. For example, in Spanish, exclamation and question marks are in the end of a sentence as common in other languages but there is also an inverted mark in the beginning of the sentence. Moreover, French adds an extra space before the question and exclamation marks whereas for example Spanish and English do not. The disparity between the punctuation conventions in different languages makes language-independent preprocessing very difficult. Thus, often all punctuation is cleaned from text to get it to a standard form, even though punctuation could carry important information.

Some of the most frequently used punctuation marks are quotation marks. The use of quotes vary, e.g. Finnish prefers the double dumb quote (") instead of the single quote more common in English ('). English uses separate opening quotation mark (‘) together with the closening one (’). Some styles and languages use subscript quotation marks, such as German quotation marks („die Sonne") or angle quotation marks with spaces (« le soleil ») used e.g., in French. Each of these have to be parsed from a multilingual text, depending on the application.

Different languages and genres have different usages for hyphens (-), whether there are en-dashes (–) or longer em-dashes (—) and whether they have space around them or not. Hyphens may be used even as marks for quotes. Apostrophes (') are used in some languages as parts of words, whereas colons (:) have the same function in some other languages. Even numbers have different spellings: sometimes there is a comma as a thousand separator (200,000 or 200'000), sometimes a space (199 999) and sometimes nothing. The decimal separator may be a period or a comma. Further examples are the rich variations in dates, times and word capitalization conventions.

3.3.3 *Document representation*

In this thesis, a *document* means a text document as understood in standard language, or a larger piece of text, a short Twitter message, a sentence, or only a phrase. A document is an independent unit that has internal structure. Usually it is assumed that a document is about one topic or is generated from a set of topics (Blei *et al.*, 2003), and other documents are more or less different from it.

Document representation means the contents of a text document transformed to a data matrix. Depending on the target application, different features from the documents can be collected. The document representation systems which are discussed below usually assume that word boundaries are known – white space between two sequences of characters is a very good indication of a word boundary. However, some languages like Chinese and Japanese do not separate words with white space and thus an additional preprocessing phase for word boundary detection may be needed for them. While document representations are typically based on the words in a document, also phrases, morphemes and characters are commonly used.

WORD–DOCUMENT MATRIX A word–document matrix is one that collects all words in a document collection into a *vocabulary* or term set $\mathcal{T}$ and lists which words exist in which documents. Table 5 shows an example word–document matrix. The frequencies of five context words are counted in each document. It can be seen that Doc 1 apparently speaks about domestic animals, Doc 2 about something that relates to a president, and Doc 3 about something else. Very common words 'the' and 'two' exist in all the documents. The matrix could also be binary: with 0's marking non-existing words and 1's for all words which appear at least once in a document. A closely related approach, a word–word matrix, in which the contexts of target words are analyzed instead of target documents, was described in Section 2.2.4.

Table 5: An example word–document matrix with three documents and five context words.

	cat	dog	president	the	two	...
Doc 1	5	3	0	25	3	
Doc 2	0	0	12	48	4	...
Doc 3	0	0	0	23	2	
⋮						

SLIDING WINDOW A sliding window approach is usually used in applications that treat the whole data set as a single document, but can be used for a set of documents also. In this approach, the 'document' or the context for each target word consists of n words before and after the target word. The collected words may be taken as a bag-of-words or the word ordering may be preserved. These are discussed below.

BAG-OF-WORDS The bag-of-words representation takes all words in a window, for example, a fixed-size window of 1–100 consecutive words, or all words in each sentence or document, into one 'bag' in which the word order is dismissed. The bag-of-words assumption is very commonly used in the NLP field due to its simplicity.

WORD ORDER Instead of using the bag-of-words assumption, the word order can be taken into account in different kinds of systems. For example, Honkela *et al.* (1995) and Schütze (1997) used the left and right contexts of a word.

N-GRAMS The use of n-grams is another way to take the word order into account. Unigrams (1-grams) correspond to words, bigrams (2-grams) are compounds of two consecutive words, trigrams (3-grams) for three words and so on. They are widely used in many NLP applications, giving keyphrase extraction as an example (Publication I). Instead of words, n-grams can also be collected for characters. Character n-grams give good information, for example, in the language identification task (Damashek, 1995). The applicability of character n-grams independently of the language was presented by Damashek (1995) in-

cluding language identification, document clustering, document classification, and exemplar-based document retrieval.

3.3.4 *Dimensionality reduction*

The dimensionality of text data matrices, such as the word–document matrix defined above, is usually very large due to the large amount of low-frequency vocabulary. The matrices are noisy containing a lot of very rare terms, possible misspellings, etc., and may contain redundant information. To decrease the matrix size and thus the computational load, and to simplify the data, dimensionality reduction is a good option. *Dimensionality reduction* is the transformation of a data matrix into a more compact, low-dimensional representation. However, if the dimensionality is reduced too much, also important information would be lost, which would make the problem harder to solve.

Many machine learning methods assume some well-known distributions, such as Gaussian or uniform distribution for document data, but the distribution of word occurrences in any language is very sparse and skewed. Dimensionality reduction and other preprocessing steps clean noise in the text data, and also move the data distribution closer to the assumed one.

Dimensionality reduction approaches can be roughly divided into two paradigms: feature selection and feature extraction (see, e.g., Sebastiani, 2002). Feature selection aims to choose the most representative features in the original feature set, whereas feature extraction creates a new set of features that are a combination of the original features.

FEATURE SELECTION The feature selection task is to choose a subset of features so that they retain as much of the important information as possible. In text analysis, it is common to apply heuristic preprocessing, such as excluding very frequent words and very rare words, because they do not give information about the word meaning through their co-occurrences with other words. This filtering can be done independently of the language using the word count information (like in, e.g., Görnerup and Karlgren, 2010), but often language-specific stop word lists are used for locating the semantically not interesting most frequent words. Filtering out stop words does not actually increase the performance of vector space models in English synonym tests, but decreases the size of the data significantly and thus speeds up the computation (Bullinaria and Levy, 2012). Other common feature selection heuristics are to remove punctuation and other non-alphabetic characters. Also more sophisticated information-theoretic methods exist (Sebastiani, 2002).

A simple algorithm, the *forward feature selection* algorithm, starts from an empty set of features and adds one feature at a time, choosing the feature which most improves an evaluation criterion. The evaluation criterion can be either a general method of selecting features (e.g., information theoretic measures) or a specific machine learning method that will ultimately be used, e.g., for supervised classification. Feature selection approaches are very promising if it is possible to evaluate the selection results (Alpaydin, 2010). This means basically labelled training data. Also unsupervised approaches for feature selection have

been proposed (e.g., Mitra *et al.*, 2002; Dy and Brodley, 2004). In this dissertation, a keyphrase extraction method has been used for unsupervised feature selection (Publications III, VII).

FEATURE EXTRACTION Feature extraction is a reparametrization task in which a small number of new features are created as combinations of the original features and thus the dimensionality is reduced. There are many methods to do feature extraction. One of the most commonly used heuristic methods in NLP is to use stemming or lemmatization. They combine the original features by merging features that correspond to the same word stem or lemma, respectively. However, stemming and lemmatization did not give significant performance advantage for English in semantic tasks (Bullinaria and Levy, 2012), but may be more important in agglutinative languages like Finnish and Turkish. Besides rule-based stemmers and lemmatizers, also statistical stemming methods exist (Creutz and Lagus, 2007; Bernhard, 2008). They can be applied in a language-independent manner.

One of the best-performing methods for feature extraction and dimensionality reduction (Deerwester *et al.*, 1990; Bingham and Mannila, 2001), also outside NLP tasks, is *singular value decomposition* (SVD). It is the optimal linear solution when the goal is to obtain the Euclidean distances between data items in the original feature space to the reduced space (Publication VI). The original data matrix $\mathbf{X}$ is decomposed as $\mathbf{X} = \mathbf{V}\Sigma\mathbf{U}^{\mathrm{T}}$, where $\mathbf{U}$ and $\mathbf{V}$ are unitary matrices and Σ is a rectangular diagonal matrix. The new data matrix in the reduced space is $\mathbf{X}_D = \mathbf{W}_D^T\mathbf{X}$, where $\mathbf{W}_D$ is formed by choosing the D column vectors of the matrix $\mathbf{V}$ corresponding to the largest singular values of $\mathbf{X}$.

The use of SVD for text document data (Benzécri, 1973) is often referred to as *latent semantic analysis* (LSA) or *latent semantic indexing* (LSI) (Deerwester *et al.*, 1990). A probabilistic version (PLSA) (Hofmann, 1999) models each word in a document as a sample from a mixture model, and different words in a document may be generated from different topics.

A common application for SVD is to calculate *principal component analysis* (PCA), that is, the projection of $\mathbf{X}$ into the space spanned by the orthogonal components of the largest variance. In PCA, the D largest eigenvalues are computed for the data covariance matrix, and the corresponding eigenvectors are chosen as the principal components of the data. An alternative approach to calculate the principal components is to use a linear neuron model (Oja, 1982). The computation of the covariance matrix involves centering the data matrix. Because text data is typically very high-dimensional and sparse, and the centering fills the sparse matrix, PCA is not commonly used for text data. If the data vectors are centered, i.e., $\mathrm{E}(\mathbf{X}) = 0$, PCA and SVD produce the same projection, but otherwise the methods are not equivalent. Another matrix factorization approach is *non-negative matrix factorization* (NMF) in which the factors $\mathbf{W}$ (basis) and $\mathbf{H}$ (encoding) are both non-negative $\mathrm{NMF}(\mathbf{X}) \approx \mathbf{WH}$. There are many algorithms to find the factors; one is a multiplicative update method (Lee and Seung, 1999).

Random projection (Ritter and Kohonen, 1989) is a computationally light method for dimensionality reduction. The data is projected with random vectors that are nearly orthogonal if the target space has sufficiently high dimen-

sion. The approach has been also referred to as *random mapping* (Kaski, 1998) and *random indexing* (Kanerva *et al.*, 2000; Karlgren and Sahlgren, 2001).

3.3.5 *Weighting and normalization*

Frequent words get a large weight in a plain word–document matrix, which disturbs, for instance, clustering text according to the topics. *Term weighting* can be used for emphasizing semantically important terms in documents. *Normalization* helps process for example documents of varying lengths. Weighting and normalization of data, e.g., word frequencies, may be used both before dimensionality reduction and after it.

The weighting schemes can be divided into local and global weighting. Local weights are counted separately for each document in a collection, whereas global weights are calculated from the whole document collection. Local and global weights can be combined with a product. For a textbook description, cf., for example, Manning and Schütze (1999) or Manning *et al.* (2008).

LOCAL WEIGHTING To perform local weighting within a document, *term frequency tf* (count of terms t_i) is a widely-used measure. The highly skewed distribution of term frequencies is often dampened using a logarithmic term frequency. These measures are shown in Table 6. Another way to produce local weighting for a word–document matrix is to apply binary weights by using ones for all non-zero entries in the matrix.

GLOBAL WEIGHTING Global weights are counted for the whole document collection. Some global weights are listed in Table 6. *Document frequency* $d(t_i)$ is the count of documents in which term t_i exists. *Inverse document frequency idf* assigns a high weight to terms that occur in few documents and that thus have been used for a specific topic only. Logarithmic *idf* is often used for dampening the effect of the weight (Jones, 1972). Also a square root of *idf* can be used.

The *global frequency* of term t_i in the whole collection is used both in entropy weighting and variance normalization. *Entropy weighting* assigns a small weight to terms that are uniformly distributed over the documents, and a large weight to terms that are concentrated in a few documents (Dumais, 1991). A common global weighting scheme for non-discrete data is to normalize the variances of the features to one. A widely used combination of the local and global weighting is *term frequency – inverse document frequency tf-idf*, in which *idf* is dampened with logarithm.

LENGTH NORMALIZATION Since documents are of different length, the total weights of document vectors change, and thus have a strong effect on e.g., document clustering. The effect of the document length can be normalized using, e.g., l_1 or l_2 norms. If using cosine distance as the similarity measure, there is no need to normalize vector lengths (Salton and Buckley, 1988).

Table 6: Local and global weighting schemes for term t_i in document j. N is the number of documents.

Weighting	
Local weightings	
Term frequency (*tf*)	$c_j(t_i)$
Logarithmic *tf*	$\log(1 + c_j(t_i))$
Global weightings	
Document frequency (*df*)	$d(t_i)$
Global frequency $G(t_i)$	$\sum_j c_j(t_i)$
Inverse document frequency (*idf*)	$\frac{N}{d(t_i)}$
Logarithmic *idf*	$\log \frac{N}{d(t_i)}$
Square root *idf*	$\sqrt{\frac{N}{d(t_i)}}$
Entropy weighting	$1 - \sum_j \frac{p_{ij} \log p_{ij}}{\log N}$, where $p_{ij} = \frac{c_j(t_i)}{G(t_i)}$
Variance normalization	$\sigma_{t_i}^{-1} = (\frac{1}{N-1} \sum_j (c_j(t_i) - \frac{G(t_i)}{N})^2)^{-\frac{1}{2}}$
Combinations	
tf-idf	$c_j(t_i) \log \frac{N}{d(t_i)}$

3.3.6 *Language models*

Language models measure the probability of generating a certain string. The n-gram language models consider sequences of n consecutive words: the *unigram language model* estimates each word independently, whereas the *bigram model* conditions on the previous word. For example, given four words t_1, t_2, t_3, t_4 the unigram probability of the phrase is $P_{uni} = P(t_1)P(t_2)P(t_3)P(t_4)$ and the bigram probability is $P_{bi} = P(t_1)P(t_2|t_1)P(t_3|t_2)P(t_4|t_3)$. Also longer n-grams and more advanced models can be used. Language models are used in various NLP tasks, including information retrieval (Manning *et al.*, 2008), part-of-speech tagging (Goldwater and Griffiths, 2007) and machine translation (Koehn, 2010). They help in choosing from a list of alternatives the sequences of words that are most probable in a language. Language models have also been applied to domain adaptation in machine translation by training the translation system with a bilingual corpus in one domain and training the language models in the target domain (Koehn and Schroeder, 2007; Dobrinkat and Väyrynen, 2010).

3.4 CLUSTERING METHODS

Clustering is an unsupervised learning approach to group similar data items together. Clustering methods aim to minimize distances inside a cluster and maximize distances between clusters. Clustering methods can be divided in hard clustering, in which a data item belongs to exactly one cluster, and soft

clustering, in which a data item may belong to several clusters with a certain degree.

Another division of clustering methods can be done according to the clustering result: hierarchical and non-hierarchical clustering. The latter is also called partitional or flat clustering. An essential procedure in clustering is to calculate distances between data items. In the following, distance measures are first discussed and then different clustering approaches are presented for textual data.

3.4.1 *Distance measures*

Clustering methods measure the distances, or similarities, between data items. Many of them use a specific distance measure. A *distance measure* is defined here as a non-negative, symmetric and reflexive function. Some distance measures satisfy also stricter conditions, for example, triangle inequality, which thus satisfies the properties of a metric. Some distance measures are listed in Table 7. They are defined for two data vectors $\mathbf{x}_i = [x_{1,i}, \dots x_{n,i}]^{\mathrm{T}}$ and $\mathbf{x}_j = [x_{1,j}, \dots x_{n,j}]^{\mathrm{T}}$.

Table 7: Distance measures between column vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ of the data matrix $\mathbf{X}$. Partly from Publication V.

Measure	**Distance**	**Measure**	**Distance**
Euclidean	$\sqrt{(\mathbf{x}_i - \mathbf{x}_j)^{\mathrm{T}}(\mathbf{x}_i - \mathbf{x}_j)}$	Cosine	$1 - \frac{\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_j}{\|\mathbf{x}_i\|_2 \|\mathbf{x}_j\|_2}$
Standardized Euclidean[1]	$\sqrt{(\mathbf{x}_i - \mathbf{x}_j)^{\mathrm{T}}\mathbf{V}^{-1}(\mathbf{x}_i - \mathbf{x}_j)}$	Correlation[2]	$1 - \frac{(\mathbf{x}_i - \bar{\mathbf{x}}_i)^{\mathrm{T}}(\mathbf{x}_j - \bar{\mathbf{x}}_j)}{\|\mathbf{x}_i - \bar{\mathbf{x}}_i\|_2 \|\mathbf{x}_j - \bar{\mathbf{x}}_j\|_2}$
Mahalanobis[3]	$\sqrt{(\mathbf{x}_i - \mathbf{x}_j)^{\mathrm{T}}\mathbf{C}^{-1}(\mathbf{x}_i - \mathbf{x}_j)}$	Spearman[4]	$1 - \frac{(\mathbf{r}_i - \bar{\mathbf{r}})^{\mathrm{T}}(\mathbf{r}_j - \bar{\mathbf{r}})}{\|\mathbf{r}_i - \bar{\mathbf{r}}\|_2 \|\mathbf{r}_j - \bar{\mathbf{r}}\|_2}$
Squared Euclidean	$\sum_{k=1}^{n} (x_{k,i} - x_{k,j})^2$	Bray-Curtis	$\frac{\sum_{k=1}^{n} \|x_{k,i} - x_{k,j}\|}{\sum_{k=1}^{n} (x_{k,i} + x_{k,j})}$
City block	$\sum_{k=1}^{n} \|x_{k,i} - x_{k,j}\|$	Bray-Curtis 2	$\frac{\sum_{k=1}^{n} \|x_{k,i} - x_{k,j}\|}{\sum_{k=1}^{n} (\|x_{k,i}\| + \|x_{k,j}\|)}$
Chebychev	$\max_{1 \le k \le n} \{\|x_{k,i} - x_{k,j}\|\}$	Canberra[5]	$\sum_k \frac{\|x_{k,i} - x_{k,j}\|}{\|x_{k,i}\| + \|x_{k,j}\|}$

[1] $\mathbf{V}$ is a $n \times n$ diagonal matrix of variance of the k^{th} variable on its k^{th} diagonal element

[2] $\bar{\mathbf{x}}_i$ is the mean vector of elements $\mathbf{x}_i$

[3] $\mathbf{C}$ is the data covariance matrix

[4] $\mathbf{r}_i$ is the coordinate-wise rank vector of $\mathbf{x}_i$ and $\bar{\mathbf{r}}$ contains mean ranks of an n-dimensional vector, i.e., $(n+1)/2$

[5] The sum is taken over those k for which $|x_{k,i}| + |x_{k,j}| \neq 0$

City block, Euclidean, and Chebychev distances are the standard special cases of the l_p-metric (Deza and Deza, 2009). The standardized Euclidean and Mahalanobis distances can be reduced to the Euclidean distance. Cosine and corre-

lation measure the angle between two data vectors. The correlation distance is based on the formulation of Pearson's correlation coefficient between two variables. Spearman distance is based on Spearman's rank correlation coefficient. Bray-Curtis and Canberra distances are less conventional in the NLP domain. They originate from ecological and environmental research (Clarke *et al.*, 2006). The Bray-Curtis distance is for non-negative data and Bray-Curtis 2, a modified version which uses absolute values, was presented in Publication V.

Other widely used distance measures in text data clustering are Jaccard coefficient, Pearson correlation and graph distance measures (Huang, 2008; Madylova and Öğüdücü, 2009; Schenker *et al.*, 2003; Strehl *et al.*, 2000).

3.4.2 *Non-hierarchical clustering*

Non-hierarchical or partitioning clustering algorithms divide data into dissimilar groups. The methods end up with flat clustering of the data.

K-MEANS *K-means* is a special case of the *expectation maximization* (EM)-based clustering algorithm (Bishop, 2006, pp. 436–437; Kanungo *et al.*, 2002). It is a simple and efficient hard clustering algorithm that clusters data items into k clusters which are represented with cluster centroids. The algorithm starts with a random initialization and then alternates between two steps: assignment of data items to their nearest cluster centroids, and updating the centroids to the means of the data items assigned to the clusters. Different distance measures can be used while the Euclidean distance metric is a common choice. More formally, for a set of data vectors $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \subset \mathbb{R}^D$ the *k-means* algorithm tries to find k cluster centroids $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_k\} \subset \mathbb{R}^D$ and a partition of the data set $S = \{S_1, S_2, \ldots, S_k\}$ such that a cost function is minimized with respect to a given distance measure. The *k-means* algorithm using the Euclidean distance finds spherical clusters. By using other distance measures the assumed cluster shapes are different, for example, standardized Euclidean finds elliptical and city block cubical clusters.

K-means is an effective algorithm but very sensitive to the initialization. Thus, often results are shown as an average of several runs. *K-means* can be calculated iteratively which leads to fast convergence. However, the algorithm stops easily to a local optimum (Theodoridis and Koutroumbas, 2008). Many modifications to *k-means* have been proposed, such as *harmony k-means clustering* (Mahdavi *et al.*, 2008) and *model-based k-means* (Zhong and Ghosh, 2005).

SELF-ORGANIZING MAP The *self-organising map* (SOM) (Kohonen, 2001) is an artificial neural network that has been used for data clustering, dimensionality reduction and information visualization. The SOM fits prototype vectors to the data items. Usually the prototype vectors are organized in a two-dimensional lattice, forming a so-called map. During the unsupervised training, the prototype vectors will start to approximate the data distribution, and the prototype vectors will self-organise so that neighbouring prototypes will model mutually similar data points. The resulting map is organized in a way that similar input data vectors locate close to each other on the map. The SOM has been

applied to textual data, such as clustering document collections (Kaski *et al.*, 1998) and part-of-speech clusters (Honkela *et al.*, 1995).

3.4.3 *Hierarchical clustering*

In order to show the hierarchical structure of a data set, *hierarchical clustering* can be used to build a clustering tree, a *dendrogram*, out of the data. Hierarchical clustering methods can be divided into agglomerative, divisive and hybrid methods. *Agglomerative* methods put each data item into a separate cluster and start combining the most similar clusters. *Divisive* methods begin with a single cluster and start dividing it. These methods use a linkage criterion how the data items are combined: single-link, in which the shortest distance between two clusters defines the similarity between the clusters; complete-link, in which the similarity is calculated between the two most-dissimilar members; and group-average, a combination of the first two criteria.

Repeated cluster bisectioning approach (Zhao and Karypis, 2004) can be used for getting a hierarchical solution from a non-hierarchical clustering method, such as *k-means*. This approach and a set of cosine-based similarity measures achieved better clustering performance than agglomerative clustering (Zhao and Karypis, 2005). Within the agglomerative clustering methods, group-average linkage performed best.

3.4.4 *Graph-based clustering*

A graph representation is built from a similarity matrix which contains pointwise similarities between data points. The graph has the data points as vertices and the similarities as weights of the edges between two vertices. Clustering within a graph may be divisive or agglomerative, similarly to the hierarchical clustering case. Most of the proposed algorithms are divisive (Chen and Ji, 2010). *Spectral clustering* is a divisive graph-based clustering algorithm that is shown to outperform *k-means* (von Luxburg, 2007). Graph-based clustering algorithms have been applied to various NLP tasks, such as document clustering (Schenker *et al.*, 2003) and creation of word similarity networks (Görnerup and Karlgren, 2010).

3.5 DENSITY ESTIMATION METHODS

Another set of unsupervised methods estimate underlying probability density functions based on observed data. *Probabilistic generative models* create a model for both the distribution of the data and the output. With those models, it is possible to sample from the distributions to get synthetic data points (Bishop, 2006). A simple *likelihood function* gives the probability for data, given a model parametrization (Manning and Schütze, 1999). To take a derivative easier, the logarithm of the likelihood, the *log-likelihood function*, is often used because it achieves the maximum value at the same point as the likelihood function. The

maximum likelihood estimate (MLE) estimates the parameters of a statistical model, maximizing the likelihood function.

3.5.1 *Latent variables*

One of the most well-known methods for latent variable estimation is *principal component analysis* (PCA) (Hotelling, 1933), discussed earlier in Section 3.3.4 as a method for feature extraction and dimensionality reduction. While PCA finds correlations in a single set of random variables, *canonical correlation analysis* (CCA) (Hotelling, 1936) compares two sets of variables. It finds linear projections for each set of variables so that the correlation between the projections is maximized (Borga, 1998; Bach and Jordan, 2003; Hardoon *et al.*, 2004). For each variable pair $\mathbf{x}$ and $\mathbf{y}$, the goal is to find linear transformations into scalars, $u_1 = \mathbf{a}^{\mathrm{T}}\mathbf{x}$ and $v_1 = \mathbf{b}^{\mathrm{T}}\mathbf{y}$, so that the correlation between the scalars is maximized:

$$\rho = \max_{\mathbf{a},\mathbf{b}} \mathrm{corr}(u_1, v_1) = \max_{\mathbf{a},\mathbf{b}} \frac{E[\mathbf{a}^{\mathrm{T}}\mathbf{x}\mathbf{y}^{\mathrm{T}}\mathbf{b}]}{\sqrt{E[\mathbf{a}^{\mathrm{T}}\mathbf{x}\mathbf{x}^{\mathrm{T}}\mathbf{a}]E[\mathbf{b}^{\mathrm{T}}\mathbf{y}\mathbf{y}^{\mathrm{T}}\mathbf{b}]}}. \tag{9}$$

CCA has been used for language data in document representation evaluation (Publication VI), cross-lingual mate retrieval (Vinokourov *et al.*, 2003) and learning bilingual lexicons from two comparable monolingual corpora (Haghighi *et al.*, 2008).

PCA can be interpreted as a linear Gaussian model, and a related but non-Gaussian model is *independent component analysis* (ICA) (Jutten and Hérault, 1991; Comon, 1994; Hyvärinen *et al.*, 2001). In ICA, the observed data variables $\mathbf{x}$ are given as a linear combination of statistically independent latent variables $\mathbf{x} = \mathbf{A}\mathbf{s}$. One efficient method to calculate both the independent components $\mathbf{x}$ and the mixing matrix $\mathbf{A}$ is to use the *FastICA* algorithm (Hyvärinen and Oja, 1997). ICA has been used in many NLP approaches, such as *WordICA* for discovering linguistic categories (Honkela *et al.*, 2010). ICA can also be used for solving the *blind source separation* problem (Cardoso, 1998).

3.5.2 *Graph-based models*

Graphical models consist of vertices connected by edges, as discussed in Section 3.4.4 from clustering point of view. The *Markov* model can be applied to sequential observations. In a first-order Markov chain, each observation is conditioned on the value of the previous observation. In a second-order chain the observation depends on two previous observations, etc. (Markov, 1913; Bishop, 2006). The *hidden Markov model* (HMM) (Baum *et al.*, 1970) is a generative model that assumes latent variables, each of which conditions one of the observed variables. HMM has been used for a range of NLP tasks, including part-of-speech tagging (Goldwater and Griffiths, 2007) and named entity recognition (Zhao, 2004). *Latent Dirichlet allocation* (LDA) (Blei *et al.*, 2003) is a generative probabilistic Bayesian model, in which each word in a collection is modelled as a finite mixture over a set of underlying topics. The topics in turn are

modelled as an infinite mixture over a set of underlying topic probabilities. LDA has been used for document classification (Blei *et al.*, 2003), ad-hoc retrieval (Wei and Croft, 2006) and image annotation (Feng and Lapata, 2008).

3.5.3 *Kernel methods*

Usually word data matrices are so large that the first idea is to reduce the dimensionality to be able to do any further calculations. In contrast, *kernel methods* map the original space into a higher-dimensional space where simple linear operations are able to find complex patterns which are non-linear in the original space. A textbook description of kernel methods, also for text data, can be found in Shawe-Taylor and Cristianini (2004).

3.6 EVALUATION

Evaluation of NLP methods is a problem itself. Since natural language is highly redundant, associative and context-dependent, usually there is not a clear best solution for a task. Typically, NLP problems are not well-defined and even humans have different opinions about the correct answer for a problem, for example, in document clustering or machine translation: usually there are many possible answers and their ranking depends on the subjective preferences of human beings. However, some NLP tasks, mostly based on linguistic theories, have a well-defined target output that is easy to agree between humans, such as part-of-speech tagging and morpheme segmentation. Correctness is one criterion to be evaluated, other criteria include the speed of the system, memory requirements, integrability to other systems and customization to the users' needs (Koehn, 2011).

Machine learning systems need training data, and evaluation is run for a separate test set. If the system parameters require tuning, another separate data set, a development set, is used. The generalization skills of machine learning systems to unseen data decrease if the system overfits to the data. Because of this, the size of the training corpus should be proportional to the number of features (Sebastiani, 2002). In evaluation, the results of a machine learning method are compared to a *baseline*, which may be a more simple method, such as classifying all data items to the most common class in the data set in a classification task, or the current *state of the art* method in the field.

3.6.1 *Evaluation approaches*

The evaluation methods for NLP can be divided into a few evaluation paradigms. A commonly used paradigm is *manual evaluation* in which a human rates the goodness or correctness of the system output. This approach raises the problem of subjective preferences in the case of using only one domain expert for evaluation. This disadvantage is alleviated by using several human evaluators (Jones and Paynter, 2001). There are many ways to combine the differing opinions of the evaluators, for instance, measuring the inter-evaluator

agreement between multiple evaluators using the *kappa coefficient* (Radev *et al.*, 2003). Manual evaluation can also be performed in a task, such as in information retrieval (Mani and Bloedorn, 1999).

Another approach is *tagging* or *manual labelling* of a test data set. The labels give the correct classification answers for evaluation. There is a large difference in the performances between the evaluations run with human-labelled test data sets and human-rated system outputs. For example, in keyphrase extraction, humans tend to give good points for extracted keyphrases (Jones and Paynter, 2001) but if the humans extract an evaluation list of keyphrases beforehand from the original documents and the automatically extracted keyphrases are compared to them, the performance is much lower (Kim *et al.*, 2010b). Another aspect is the load of work: labelling a test data set needs to be done only once but rating the output needs to be done for every run of the NLP system.

The evaluation approaches above require a substantial amount of manual work. *Automatic evaluation* typically compares the performance of a system to manually labelled resources. Many different kinds of resources can be used: dictionaries (Rapp, 1999), ontologies (Miller, 1995; Fellbaum, 1998), many pieces of meta information in Wikipedia (Publication I), and parallel corpora (Vinokourov *et al.*, 2003). Also author-provided information, such as keywords and topic categories can be used in evaluation. These kinds of sets are called the *gold standard* or *ground truth*. Even otherwise fully automatic NLP methods typically use manually created evaluation sets. However, automatic evaluation measures, such as the *BLEU* measure used in machine translation, may not correspond well with human evaluations (Koehn, 2010).

One way to categorize automatic evaluation methods is to divide them into direct and indirect evaluation methods. The *direct* methods evaluate the quality of the system outputs, as discussed above. In *indirect* evaluation, the evaluation is conducted in an external application, that is, the output of the system is used as an input of a task which may be easier to evaluate.

3.6.2 *Evaluation measures*

While the correct answers, i.e., labels or tags, for each data item are available, the most commonly used evaluation measures are precision, recall and F-measure. *Precision P* measures the number of correct answers compared to all the answers the system gives. *Recall R* measures the number of correct answers compared to all the possible correct answers for the task. As an example, consider a classification task of automatically classifying hybrid cars from other cars in an imaginary data set. A contingency table (Manning *et al.*, 2008, p. 155) about true and false positives and negatives can be drawn, as shown in Table 8.

Table 8: Contingency table of classifying cars as hybrid or not.

	Hybrid	**Not hybrid**
Classified as hybrid	true positives (tp)	false positives (fp)
Classified as not hybrid	false negatives (fn)	true negatives (tn)

Precision P and recall R can be given as

$$P = \frac{\text{tp}}{\text{tp} + \text{fp}}, \tag{10}$$

$$R = \frac{\text{tp}}{\text{tp} + \text{fn}}. \tag{11}$$

F-measure combines precision and recall as a harmonic mean, giving one evaluation value instead of two:

$$F = 2\frac{P \cdot R}{P + R}. \tag{12}$$

Usually when tuning the system, it is difficult to obtain a higher F-measure: the higher the precision gets, the lower the recall falls and vice versa.

The measure of *accuracy* A involves also the true negatives $A = (\text{tp} + \text{tn})/(\text{tp} + \text{fp} + \text{tn} + \text{fn})$ but is not suitable for problems in which the amount of negative samples is much larger than positive, for example in information retrieval problems (Manning *et al.*, 2008, p. 143).

In the case when the order of several results is important, such as in IR, measures dealing with ranked results are needed. Some commonly used measures are *precision at k* and *mean average precision* (MAP) (Manning *et al.*, 2008, pp. 147–148). In clustering, *purity* and *entropy* may be used (Strehl *et al.*, 2000). Also measures for specific NLP tasks have been proposed (e.g., Virpioja *et al.*, 2011b; Publication VI).

3.6.3 *Statistical significance*

To be able to tell whether the difference between the results of two systems is significant, *statistical significance* tests are used. They measure the confidence that the difference between two results did not happen just by chance. The idea is to define a *null hypothesis* H_0 that the samples come from the same distribution (and there is no statistical difference) and test, whether the null hypothesis can be rejected. The commonly used significance levels are 0.05, 0.01 or 0.005, which define the probability that the tested significance anyway does not hold.

There are various tests for different usages. Some commonly used tests are the *Student's t-test*, *Welch's t-test* (applied in Publication VII), *Wilcoxon signed rank test* (applied in Publications IV and VI), *Pearson's* χ^2 *(chi-square) test* and *log-likelihood ratio*. Typically, t-tests assume normal distribution which does not usually hold for language data. Also χ^2 test can be counted with normal distribution but the χ^2 distribution is another option. Within the natural language processing, statistical tests have been used as evaluation measures but also, for example, in collocation identification (Thanopoulos *et al.*, 2002), identification of translation pairs (Church and Gale, 1991), comparing corpora (Rayson *et al.*, 2004) and keyword identification (Scott, 2001; Sharoff, 2010).

4 LANGUAGE-INDEPENDENT APPLICATIONS

In this chapter, the main contributions of this dissertation are summarized. The contributions are to the field of natural language processing, considering fully automatic approaches, language-independent methods and subjectivity in language use. The proposed methods and approaches are presented in this chapter in the context of other language-independent approaches in the literature. While doing that, it is also shown how the machine learning methods and preprocessing discussed in Chapter 3 can be applied to different NLP tasks to obtain language-independent methods. The first task presented in this chapter, keyphrase extraction (Section 4.1), belongs to a group of NLP tasks called *information extraction*. In information extraction, knowledge is collected from natural language data, such as document collections. Besides keyphrase extraction, information extraction includes for example named entity recognition, sentiment analysis and document summarization, which are discussed further in Section 4.1.4.

The next topic of the chapter, *taxonomy learning* in Section 4.2, applies hierarchical clustering to document feature vectors. In the *lexical choice* task in Section 4.3, a range of flat clustering methods are tested on a large set of linguistic features. In Section 4.4, two topics in creating document representations are discussed. The first topic is document clustering with the *k-means* method. The second topic is an evaluation method for representations of text documents. One of the main themes of this dissertation, *subjectivity*, is considered in Section 4.5: An approach for user modelling in text difficulty assessment is presented. In Section 4.6, two other major NLP tasks that can be approached with language-independent methods, information retrieval and machine translation, are discussed.

4.1 KEYPHRASE EXTRACTION

Automatic *keyphrase extraction* is an information extraction task in which the content of a document is represented with a few *keywords* or phrases known as *keyphrases*. Keyphrases are supposed to be available in the processed documents themselves, and no additional lists of potential keyphrases are needed. Keyphrase extraction is a text mining procedure that can be used as a basis for other, more sophisticated text analysis tasks, such as information retrieval (Gutwin *et al.*, 1999), text summarization (D'Avanzo, 2005), document clustering (Hammouda *et al.*, 2005), taxonomy learning (Publication III), invalidity search of patents (Verma and Varma, 2011), assessment of text difficulty (Publication VII), analysis of comparable corpora (Sharoff, 2010) and qualitative analysis of abstracts (Klami and Honkela, 2007). Some keyphrase extraction methods also provide scores of goodness for being a keyphrase and thus can be used similarly to traditional document weighting measures.

Within the proposed statistical keyphrase extraction methods in the literature, term frequency is naturally used as the main feature representing keyphrases. Most of the presented methods require a reference corpus or a training corpus to produce keyphrases. Some of the best-known supervised approaches, that require a labelled training set, include the *KEA* (Frank *et al.*, 1999) and *GenEx* (Turney, 2000) methods. Unsupervised and statistical keyphrase extraction methods naturally do not use labelled training sets, but most of the unsupervised approaches presented in the literature use stemming, stop word lists and part-of-speech tags (to locate noun phrases or nominal groups) in their preprocessing and thus are specific to the used language. Out of these preprocessing steps, it is usually possible to leave stemming out and replace language-specific stop word lists with frequency-based filtering of words. These would produce approaches that are more independent of the used language and still get reasonable extraction results.

El-Beltagy and Rafea (2009) used heuristic rules to extract keyphrases, such as a threshold for the first appearance of a keyphrase in a document and a ratio of single to compound terms. In spite of some language-specific components, the system was shown to work for documents both in English and Arabic. Another approach used *mutual information* and *log-likelihood* of terms and was tested for both English and Chinese (Pantel and Lin, 2001). Pasquier (2010) proposed a method exploiting sentence clustering, dimensionality reduction and *latent Dirichlet allocation* (Blei *et al.*, 2003). Damerau (1993) proposed a method that compares relative frequencies of terms in a document to a reference corpus. Approaches based on nominal groups include the use of word co-occurrence statistics (Matsuo and Ishizuka, 2004), genetic algorithms for noun phrases patterns (Wu and Agogino, 2003), noun phrase clustering in English and Japanese (Bracewell *et al.*, 2005), and graph-based methods *TextRank* (Mihalcea and Tarau, 2004) and *DegExt* (Litvak *et al.*, 2011). Also other information, such as pdf document meta information (Nguyen and Luong, 2010), has been proposed. A closely related task to keyphrase extraction is domain-specific term extraction. Vivaldi and Rodríguez (2010) proposed a language-independent approach that uses Wikipedia categories and pages.

4.1.1 *Language-independent keyphrase extraction (Publications I, II)*

Most of the keyphrase extraction methods proposed in the literature are either supervised methods which require a training set, or use language-specific components. In contrast, the language-independent keyphrase extraction method *Likey*, presented in Publication I, does not use any language-specific preprocessing or resources. It extracts keyphrases from a document based on phrase frequency ranks. The only language-specific component needed is a reference corpus in each language. Thus *Likey* is easily portable to new languages. The *Likey* method include the *Likey ratio*, a post-processing step and a reference corpus.

The Likey method

The *Likey* ratio (Publication I) compares frequency ranks of words in a document to a reference corpus and reorders the words and phrases of the document to have the best keyphrases in the beginning of the list.

THE LIKEY RATIO The *Likey* ratio (Publication I) for each phrase is defined as

$$L(t,j) = \frac{rank_j(t)}{rank_r(t)}, \tag{13}$$

where $rank_j(t)$ is the rank value of phrase t in document j, and $rank_r(t)$ is the rank value of phrase t in the reference corpus. The phrase set contains all the n-grams of the document up to phrase length n. The rank values are calculated separately for each n as the ordered frequencies of the phrases; the phrase having the largest frequency gets the rank of 1. In the case of the same frequency value the rank value also stays the same. If phrase t does not exist in the reference corpus the value of the maximum rank for the phrases of length n is used: $rank_r(t) = max\_rank_r(n) + 1$. A closely related method has been introduced by Damerau (1993), but it compares unigram frequencies instead of n-gram ranks as in our case.

The *Likey* ratio ranks all the words that occur more than once in a document. The phrases that have the smallest ratio are the best candidates for being keyphrases. The method does not use any parameters to be tuned and is applicable to documents practically in any domain and language.

However, the *Likey* ratio cannot select the number of keyphrases extracted from each document and, furthermore, it cannot be directly applied to term weighting. Thus further versions of *Likey* were presented in Publications III and VII. Length-normalized $Likey_N$ introduces a weighting scheme to automatically select the number of keyphrases depending of the length of the document. Weighted $Likey_W$ incorporates weights for the keyphrases and it can be used similarly to other term weighting schemes for document representation. The third variation, percentage $Likey_P$ simply selects certain percentage of the document words as keyphrases. These approaches have some additional parameters to be selected manually or from data and thus they are less independent of the language than the original *Likey* method.

LENGTH-NORMALIZED LIKEY To be able to automatically select a suitable number of keyphrases from documents of different length the *length-normalized Likey ratio* $Likey_N$ (Publication III) was proposed:

$$L_N(t,j) = \frac{L(t,j)}{M_j^a}, \tag{14}$$

where M_j is the number of word tokens in document j and a is a constant $a \in [0, 0.5]$. With this measure it is possible to select a common threshold value η that independently of the length of a document can be used as a threshold between good keyphrase candidates and the non-keyphrases. The length-normalized *Likey* ratio is scaled to be from 0 to 1 and thus the *Likey weight* of

the length-normalized $Likey_N$ for each keyphrase t in document j was introduced in Publication III:

$$w(t,j) = 1 - \frac{L_N(t,j)}{\eta} \quad \text{if } L_N(t,j) < \eta,$$

which is close to 1 for the best keyphrase candidates and not defined for poor candidates.

WEIGHTED LIKEY As a third version of *Likey* the weighed $Likey_W$ was proposed in Publication III. The *Likey ratio* (Equation 13) cannot be used directly as keyphrase weights since the best keyphrases get the smallest *Likey ratio* values. We thus scale the ratio to values between [0, 1], where values closer to 1 are the best keyphrases. The *Likey weight* of the weighted $Likey_W$ for phrase t in document j is calculated as in Publication III:

$$w_2(t,j) = \begin{cases} (\frac{1}{\tau} - L(t,j)) \cdot \tau & \text{if } L(t,j) < \frac{1}{\tau} \\ 0 & \text{otherwise,} \end{cases} \tag{15}$$

where $1/\tau$ is the maximum *Likey ratio* taken into account.

PERCENTAGE LIKEY In Publication VII, a new version for automatic selection of the number of keyphrases from each document was proposed. Percentage $Likey_P$ simply selects a percentage of φ terms per 100 words in a document to be keyphrases of the document. This differs from the previous variations by directly assigning the number of extracted keyphrases per 100 document words, rather than selecting a threshold for the *Likey* weight.

Post-processing

After calculating *Likey* ratios, part of the candidate keyphrases of length $n > 1$ are removed in a post-processing step. Two versions were proposed in Publications I and II.

ORIGINAL The original post-processing step was proposed in Publication I and was used also in Publications III and VII. If the reference rank value $rank_r$ of any of the single words constituting a phrase is smaller than the rank of the whole phrase, that means, one word is more common than the phrase, the phrase is removed. This procedure excludes for example phrases that contain function words such as 'of' or 'the', in a phrase 'of the world'. This uses the assumption that the maximum rank value is usually smaller for longer phrases than for single words, since the frequencies of longer phrases are lower. In addition to the removal above, also lower-rated subphrases of any already selected keyphrase are also removed, excluding e.g., 'language model' if 'unigram language model' has already been accepted as a keyphrase.

THRESHOLDED An alternative post-processing approach was proposed in Publication II for the first post-processing step: If one of the words composing a phrase has a rank of less than a threshold ξ in the reference corpus, the

phrase is removed from the keyphrase list. This is basically a statistical approach to the use of a stop word list: the procedure excludes mostly phrases with function words. The second post-processing step is the same as in the original one: phrases that are subphrases of those that have occurred earlier on the keyphrase list are removed.

Reference corpus

The reference corpus for *Likey* acts as a sample of general language. The corpus should be as large as possible to contain sufficiently many examples of the language use. The choice of the reference corpus has a substantial effect on the extraction results. For instance, keyphrase extraction from a document that belongs to a specific domain or a document of unusual style may lead to results that describe more the style than the topic of the document if the reference corpus is inapplicable.

In all the *Likey* publications of this dissertation, the Europarl corpus (Koehn, 2005) of European parliament plenary speeches was used as the reference corpus. The 11 Europarl languages and their word counts were shown in Table 2. The strength of this kind of parallel corpus is that if the analysed documents are similar, the keyphrase extraction results are comparable between languages. The other side of the coin is the biased content: most of the texts are translations; the speech-to-text transcriptions are different language from language originally in written form; the topics of the texts are concentrated on European politics and political rhetoric including over-representation of certain words ('president', 'European', 'Mr'); and there is poor representation of many common words and phrases like personal pronouns ('she' and 'him'), contractions ('I'm', 'I'll') and acronyms and abbreviations ('Wed', '*et al.*', 'ok'). A better reference corpus would be one that contains a large amount of language samples in different styles and purposes. This kind of corpus could be collected from the Internet, like Google n-gram corpus[1], but it would vary from language to language according to the amount, stylistic variation and correctness of language. In the experiments, that kind of inconsistency was not desirable and thus Europarl was used.

4.1.2 *Keyphrase evaluation*

Keyphrase evaluation is not a straightforward procedure. As is typical to natural languages in general, individuals have different opinions about a good list of keyphrases for a document. The keyphrase extraction evaluation results depend heavily on the evaluation strategy: whether author-provided keyphrases are used, or whether human evaluators assess the list of automatically extracted keyphrases. If the human evaluators provide their own lists of reference keyphrases, there is a difference between the extraction performance compared to reference keyphrases that are selected from the documents and reference keyphrases which may contain also other words than those existing in the doc-

1 http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2006T13

uments. Kim *et al.* (2010b) listed monolingual keyphrase evaluation sets that are constructed using different evaluation strategies.

In a multilingual domain, it is very difficult to get human evaluators for a multitude of languages. If there are more languages than one person knows, several parallel evaluators are needed, and the results are no longer comparable due to the subjectivity of the evaluation. Thus, an automatic evaluation would be preferred. However, as far as we were aware, there was not a large multilingual set of documents expanded with lists of keyphrases available. The commonly used evaluation measures to match the extracted keyphrases and the reference keyphrases are precision, recall, and F-measure. Kim *et al.* (2010a) proposed n-gram-based evaluation metrics to account for near-misses, that is, partly-matched keyphrases.

In this dissertation, two keyphrase evaluation settings are considered. The first setting, proposed originally in Publication I, uses manually constructed information that was originally produced for human use and is available for hundreds of languages – Wikipedia. The second evaluation setting uses a set of scientific articles.

Wikipedia links

A multilingual keyphrase evaluation set was collected from Wikipedia, the free multilingual online encyclopedia.[2] The evaluation method is presented in Publication I. The evaluation consists of 11 European languages: four Romance languages: Spanish (es), French (fr), Italian (it) and Portuguese (pt); five Germanic languages: Danish (da), German (de), English (en), Dutch (nl) and Swedish (sv); and Greek (el) and Finnish (fi).

Wikipedia articles link to other Wikipedia articles in order to give further explanations for the important content words and phrases of the article at hand. We assume that the linking practices to other articles are independent of the used language. To gather the reference keyphrases for a Wikipedia article, the links to other articles (in the same language) are collected, and from them, those articles that also link back to the original article are selected. The set of two-way linking article titles are then treated as the set of reference keyphrases for the Wikipedia article. The requirement of the backlinks remove many of those links that are definitions of semantically non-interesting words. In addition, all dates and years are removed from the obtained link collection.

To be able to compare the evaluation results between languages, articles about the same topic in each language were collected. At the time of data collection, it was quite challenging to find Wikipedia articles of adequate extent in all the 11 languages, basically due to generally quite short articles in Greek, Finnish and Danish. Finally, 10 articles having a sufficient amount of content in each of the 11 Europarl languages, in total of 110 articles, were collected for evaluation. The topics of the articles are *Beer*, *Cell (biology)*, *Che Guevara*, *Leonardo da Vinci*, *Linux*, *Paul the Apostle*, *Sun*, *Thailand*, *Vietnam War*, and *Wolfgang Amadeus Mozart*, collected in March 2008. Only the text body was taken into account and all tables and reference lists were removed. The articles constitute

2 http://wikipedia.org

a small comparable corpus in 11 languages: the articles are not translations of each other but cover the same topics. It seems that the articles in English are usually of the best quality and the articles in the other languages are more or less translations from the English version.

The set of produced evaluation keyphrases is an intersection of in-links and out-links. As an example, 67 links, the reference keyphrases, extracted from the English Wikipedia article *Cell* are listed:

> adenosine triphosphate, amino acid, anabolism, archaea, bacteria, binary fission, cell division, cell envelope, cell membrane, cell nucleus, cell theory, cell wall, centrosome, chromosome, citric acid cycle, cyanobacteria, cytoplasm, cytosol, dna, dna replication, endocytosis, endomembrane system, endoplasmic reticulum, energy, enzyme, eukaryote, extrachromosomal dna, francis crick, gene, genetic code, genetics, glucose, glycolysis, golgi apparatus, histone, hormone, human, hydrogen hypothesis, ion, last universal ancestor, life, metabolic pathway, metabolism, microfilament, mitochondrion, molecule, multicellular organism, nuclear envelope, organ (anatomy), organelle, organism, osmotic pressure, phospholipid, phosphorus, plasmid, plastid, protein, proteinoid, retrovirus, ribosome, robert hooke, signal transduction, theodor schwann, tonicity, transfection, vacuole, vesicle (biology)

Qualitatively, this link set is a very good suggestion for being a keyphrase list of the Wikipedia article *Cell*, because it contains parts of a cell (mitochondrion, vacuole, golgi apparatus, ...), where cells exist (organism, life, ...) and different cell types (archaea, bacteria, eukaryote). Further, it does not have many poorly fitting terms.

The average number of evaluation keyphrases for each language, together with the minimum and maximum counts, are shown in Figure 2. Articles in languages of less than 100 000 articles at the time of data collection (Danish and Greek) faced a problem that they contained many links to articles that were not written yet and thus did not get them to the reference keyphrase list. The number of evaluation keyphrases for Greek, Danish, Finnish, Swedish, Dutch and Portuguese is quite low, see Figure 2.

PREPROCESSING The preprocessing phase of the *Likey* experiments consists of removing punctuation and other special characters (except for within-word hyphens and colons), and replacing numbers with `<NUM>` tags. The text is split into parts according to the punctuation. The splitting ensures that n-grams do not span over multiple sentences. Punctuation for splitting text into sentences is assumed to be present in most of languages.

RESULTS In Publication I, keyphrases of length $n = 1,\ldots,4$ words were extracted. Longer phrases than four words did not occur in the output keyphrase lists in preliminary tests. As simple as *Likey* is, it produces surprisingly high-quality keyphrases in each language tested, as can be seen in Table 9. As a baseline, the state-of-the-art term weighting method that can be used also for keyphrase extraction, *tf-idf* (Salton and Buckley, 1988), was used for extracting keyphrases from the same material. *Tf-idf* is usually used for a single corpus, but in

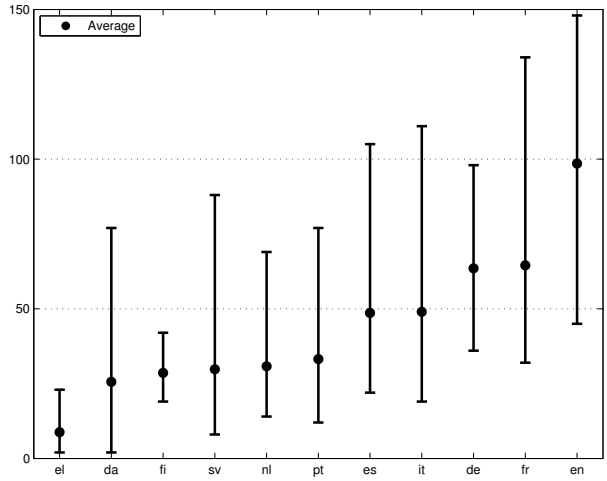


Figure 2: The average number of Wikipedia evaluation keyphrases for each language, together with the minimum and maximum evaluation keyphrase counts for each language.

this study a separate reference corpus was utilized, similarly to *Likey*. For *tf-idf*, the Europarl reference corpora were split in 'documents' of 100 sentences to be able to count the document frequencies, and the same preprocessing as for *Likey* was used. The *Likey* post-processing was applied to *tf-idf* as well. Generally, *Likey* produced longer phrases than *tf-idf*. Each keyphrase list characterised the topic quite well, and most of the extracted keyphrases recurred in every language. The *Likey* keyphrase extraction method was compared to the baseline method *tf-idf* by calculating precision and recall, according to the reference keyphrases.

For the first experiment, 60 keyphrases were extracted from each document and for the second experiment the number of keyphrases available in the reference keyphrase list for the document were extracted. The results as averages across languages are shown in Table 10, with *Likey* performing better than *tf-idf*, also if the *Likey* post-processing was added to *tf-idf*. Precision and recall values of both measures are comparatively low, but one has to take into account the nature of the evaluation set with notably varying number of reference keyphrases.

The obtained precisions and recalls of the first experiment differed significantly between languages. In Figure 3, the precision and recall of *Likey* and *tf-idf* with post-processing for each language are given. Within the 11 European languages, English and German performed best according to the precisions (*Likey:* 23.0% and 22.8%, respectively), but not that nicely according to the recalls, where Dutch and Greek performed best (*Likey:* 33.4% and 31.8%, respectively). These scores are quite understandable in the light of Figure 2: the number of reference keyphrases is over 60 on average for English and German, whereas Greek and Dutch require a smaller number of keyphrases. *Likey* and

Table 9: The first ten *Likey* and *tf-idf* keyphrases from Wikipedia articles *Cell*.

Language	***Likey* keyphrases**
da	cellens genetiske materiale, cellen, dna, organeller, prokaryoter, cellekernen, eukaryoter, cellemembranen, arvemateriale, celler
de	eukaryotischen zellen, zelle, et al, eukaryotische zellen, prokaryotische zellen, cell biol, tierischen zellen, zellkern, dna, membran
el	κύτταρο, κύτταρα, ανόργανα, υδατάνθρακες, μορίων, βακτήρια, κυττάρων, φυσικές, νερό, ουσιών
en	cell, circular dna molecule, matthias jakob schleiden, cells, molecules, membrane, eukaryotic, rna, organelles, prokaryotes
es	células, célula, celular, membrana, arn, procariotas, citoplasma, eucariotas, micras, plasmática
fi	solun, solun tukirangan, runsaasti proteiineja, aitotumallisissa soluissa, solulimassa, solukalvon, solukalvo, solut, solujen, solulima
fr	cellule, cellules, cellulaire, eucaryotes, membrane, procaryotes, l'adn, molécules, re, cellulaires
it	cellula, ribosomi reticolo endoplasmatico apparato, dna ribosomi reticolo endoplasmatico, reticolo endoplasmatico ruvido, membrana cellulare, parete cellulare, membrana cellulare nucleo, reticolo endoplasmatico liscio, cellula vegetale, cellula procariote
nl	celwand, cel, bladgroenkorrels, cytoplasma, celmembraan, bacteriën, celkern, vacuole, eukaryoten, reticulum
pt	células, célula, celular, procariontes, membrana, eucariontes, hooke, rna, dna, adn
sv	binär fission, ämnesomsättning metabolism, celler, cellen, cellens, prokaryoter, dna, eukaryota, cell, proteiner

Language	***tf-idf* keyphrases**
da	cellen, cellens, organeller, prokaryoter, cellekernen, eukaryoter, dna, cellemembranen, arvemateriale, kromosomer
de	zelle, zellen, zellkern, mitochondrien, membran, eukaryotischen, prokaryoten, eukaryoten, dna, chromosomen
el	κύτταρο, κύτταρα, ανόργανα, υδατάνθρακες, μορίων, βακτήρια, κυττάρων, και την ικανότητα, την ικανότητα, φυσικές
en	cell, cells, eukaryotic, membrane, dna, molecules, rna, organelles, prokaryotes, prokaryotic
es	células, célula, celular, membrana, micras, eucariotas, <NUM> micras, citoplasma, procariotas, arn
fi	solun, solukalvo, solulimassa, solukalvon, solut, solulima, soluissa, solujen, lihassolun, nadh:ta
fr	cellule, cellules, eucaryotes, cellulaire, procaryotes, membrane, l'adn, molécules, re, cytoplasme
it	cellula, membrana, citoplasma, mitocondri, cellulare, cellule, eucariote, reticolo endoplasmatico, endoplasmatico, ribosomi
nl	celwand, cytoplasma, bladgroenkorrels, celmembraan, cel, celkern, bacteriën, endoplasmatisch, endoplasmatisch reticulum er, reticulum
pt	células, célula, procariontes, hooke, eucariontes, membrana, celular, rna, cloroplastos, cloroplastos e
sv	cellen, cellens, prokaryoter, celler, eukaryota, metabolism, organeller, dna, prokaryota, cell

Table 10: The average precisions P and recalls R across language for *Likey*, *tf-idf* and *tf-idf* with *Likey*'s post-processing (p). N keyphrases refers to the number of reference keyphrases available for each article.

	60 keyphrases		N **keyphrases**
Method	P	R	P **&** R
Likey	0.148	0.247	0.180
tf-idf	0.123	0.220	0.138
tf-idf+ p	0.134	0.234	0.162

tf-idf performed very similarly throughout the data set, while *Likey* reached slightly better values.

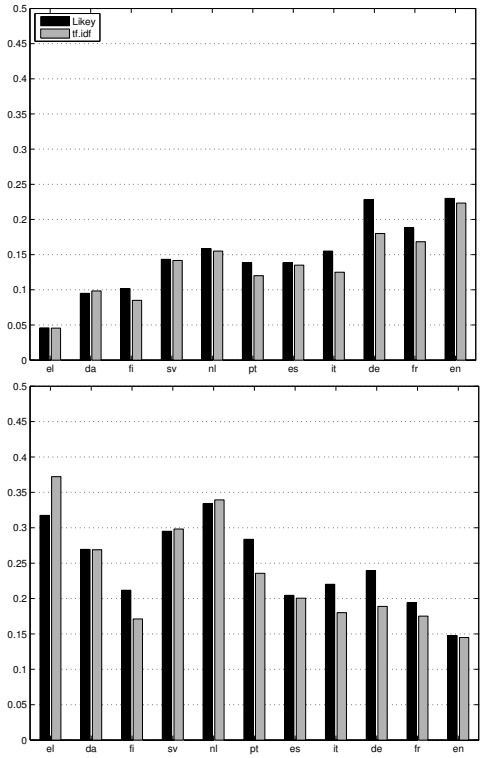


Figure 3: The average precisions (upper figure) and recalls (lower) of *Likey* (black) and *tf-idf* (grey) with post-processing for each language. The number of extracted keyphrases is 60. From Publication I.

In the second experiment, the same number of keyphrases that exists in the reference keyphrase list were extracted. The results for each language are shown in Figure 4. The average precisions are higher than those for the lists of 60 keyphrases in Figure 3. An exception is English, which seems to get most of the important keyphrases on the top 60 list.

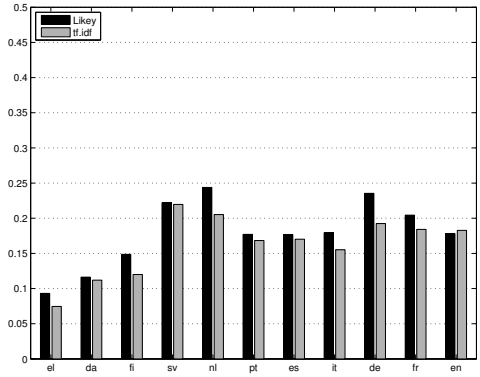


Figure 4: The average precisions of *Likey* and *tf-idf* for each language. The number of extracted keyphrases is the same as the number of keyphrases in the evaluation set.

In general, if the first three languages (Greek, Danish and Finnish) are excluded, the shape of the distribution is quite similar to both the graphs in Figure 3. *Likey* performs usually slightly better than the baseline method *tf-idf*. The evaluation corpus undoubtedly has its problems but it worked relatively well taken the difficulty of the multilingual evaluation task into account. Nevertheless, the quality of the Wikipedia articles have been and will improve.

Scientific articles

The second evaluation set for keyphrase extraction is a collection of scientific articles in English. The set is from the SemEval 2010 Challenge, Task 5, 'Automatic keyphrase extraction from scientific articles'. Since the Challenge was supervised, the data set consists of train (144 articles), trial (40 articles), and test (100 articles) sets. The length of the scientific articles in the data set is between 6 and 8 pages including tables and pictures.

Three sets of reference keyphrases were provided for the articles in each data set: reader-assigned keyphrases, author-provided keyphrases, and a combination of them. The reader-assigned keyphrases had been extracted manually from the articles, whereas the keyphrase lists provided by the authors of the articles may contain also phrases that do not exist in the articles (Kim *et al.*, 2010b). The numbers of the reference keyphrases in the data sets are shown in Table 11.

Table 11: The number of reader, author, and combined reference keyphrases for each data set.

Data set	**Reader**	**Author**	**Combined**
train	1 824	559	2 223
trial	526	149	621
test	1 204	387	1 466

PREPROCESSING In Publication II, the scientific articles were preprocessed by removing all headers, references sections, tables, figures, equations and citations. Both the scientific articles and the reference corpus (the English part of Europarl) were lowercased, punctuation was removed and the numbers were changed to `<NUM>` tags. Due to the Challenge rules, the data was stemmed with English Porter stemmer implementation provided by the Challenge organizers.

RESULTS Three different parameter settings were used with *Likey* in Publication II. *Likey-1* selected keyphrases up to three words, and *Likey-2* and *Likey-3* up to four words. The thresholded post-processing was used in all versions. The threshold value for post-processing was selected against the trial set, with $\xi = 100$ performing best. It was used for *Likey-1* and *Likey-2*. Also a bit larger threshold $\xi = 130$ was tried for *Likey-3* to exclude more function words.

The results in Publication II are shown in Table 12 for reader-assigned keyphrases and in Table 13 for the combined set of reader- and author-assigned keyphrases. The evaluation was conducted by calculating precision P, recall R and F-measure F for the top-5, 10, and 15 keyphrase candidates for each method. The baseline methods selected by the Challenge organizers were the unsupervised *tf-idf* method and the supervised *naive Bayes* (NB) and *maximum entropy* (ME) methods.

Table 12: The results for *Likey* and the baselines for the reader data set. The best precision P, recall R and F-measure F are highlighted.

	Top 5 candidates			**Top 10 candidates**			**Top 15 candidates**		
	P **%**	R **%**	F **%**	P **%**	R **%**	F **%**	P **%**	R **%**	F **%**
Likey-1	**24.6**	10.2	14.4	17.9	14.9	**16.2**	13.8	**17.2**	15.3
Likey-2	23.8	9.88	14.0	16.9	14.0	15.3	13.4	16.7	14.9
Likey-3	23.4	9.72	13.7	16.8	14.0	15.2	13.7	17.1	15.2
tf-idf	17.8	7.39	10.4	13.9	11.5	12.6	11.6	14.5	12.9
NB	16.8	6.98	9.86	13.3	11.1	12.1	11.4	14.2	12.7
ME	16.8	6.98	9.86	13.3	11.1	12.1	11.4	14.2	12.7

Table 13: The results for *Likey* and the baselines for the combined (reader+author) data set. The best precision P, recall R and F-measure F are highlighted.

	Top 5 candidates			**Top 10 candidates**			**Top 15 candidates**		
	P **%**	R **%**	F **%**	P **%**	R **%**	F **%**	P **%**	R **%**	F **%**
Likey-1	**29.2**	9.96	14.9	21.1	14.4	**17.1**	16.3	**16.7**	16.5
Likey-2	28.4	9.69	14.5	19.9	13.6	16.1	15.7	16.1	15.9
Likey-3	28.0	9.55	14.2	19.6	13.4	15.9	16.1	16.4	16.3
tf-idf	22.0	7.50	11.2	17.7	12.1	14.4	14.9	15.3	15.1
NB	21.4	7.30	10.9	17.3	11.8	14.0	14.5	14.9	14.7
ME	21.4	7.30	10.9	17.3	11.8	14.0	14.5	14.9	14.7

All *Likey* versions outperformed the baselines, while *Likey-1* performed best out of them. *Likey-1* achieved the best precision 24.6% for the top-5 candidates in the reader data set and 29.2% for the top-5 candidates in the combined data set. The best F-measure was obtained for the top-10 candidates for both reader and combined data set: 16.2% and 17.1%, respectively.

In the supervised Challenge, *Likey* was ranked as 16th out of 19 participants according to the top-15 F-measure. However, *Likey* performed much better in the beginning of the keyphrase list: according to the top-5 F-measure with both reader and author reference keyphrases the rank of *Likey* was 11.

Out of the 15 preceding systems on the ranked list, the details of 12 were published in the Challenge proceedings (Kim *et al.*, 2010b). 7 out of 12 systems were supervised approaches, including the two winning systems. Supervised approaches naturally have an advantage with their possibility to train the system with the correct keyphrases. The results of all unsupervised methods with the combined reference set are shown in Table 14. Out of the five unsupervised systems that performed better than *Likey*, only two, *KP-Miner* (El-Beltagy and Rafea, 2010) and *KX* (Pianta and Tonelli, 2010) performed better also with top-5 keyphrases. Both the systems use a language-specific list of stop words. *KP-Miner* applies also a set of constants, defined manually for the data set and domain-specific parameters about term position in the document. *KX* applies part-of-speech (PoS) information, acronym extension and has also two thresholds defined manually (like *Likey-1* does). The rest three unsupervised approaches use acronym extension and PoS tags or a lemmatizer, as well as some other additional knowledge sources (Bordea and Buitelaar, 2010; Ortiz *et al.*, 2010; Pasquier, 2010). Within the group of the unsupervised approaches, *Likey* performs on average, but when considering the language independence and simplicity, *Likey* seems very strong from that point of view.

Table 14: The results of the unsupervised approaches in the SemEval Challenge for the combined (reader+author) data set, using precision *P*, recall *R* and F-measure *F*. The system ranks (#) in the Challenge for each output list are also shown.

	Top 5 candidates				**Top 10 candidates**				**Top 15 candidates**			
	***P*%**	***R*%**	***F*%**	**#**	***P*%**	***R*%**	***F*%**	**#**	***P*%**	***R*%**	***F*%**	**#**
KP-M.	36	12	18	4	29	20	23	5	25	26	25	3
KX	34	12	17	8	27	18	22	7	24	24	24	7
Likey-1	29	10	15	11	21	14	17	14	16	17	17	16
DERIU.	27	9	14	13	23	16	19	10	22	23	22	8
UNICE	27	9	14	14	22	15	18	12	18	19	19	13
UvT	25	9	13	15	19	13	15	16	15	15	15	17
BUAP	14	5	7	18	18	12	14	17	19	19	19	11

4.1.3 *Discussion*

Likey is shown to perform better than simple baseline methods, such as *tf-idf*, for the keyphrase extraction task for both English scientific articles (Publication II) and Wikipedia articles in 11 European languages (Publication I). However, it did not rank very well in the SemEval Challenge (Kim *et al.*, 2010b), which is rather natural due to the supervised approaches, language-specific components and domain-specific features used by many of the better-ranking methods. Regardless of this, the *Likey* approach is very promising for small languages, dialects, historical versions of languages and large multilingual projects where language-specific information is not available or feasible to obtain.

Likey has been tested with 11 European languages, among which Greek and Finnish differ considerably from the other languages in the Romance and Germanic language groups. The method has given comparable results for each language and thus shows independence of the used language, at least within the Indo-European and Uralic language families. *Likey* requires only a lightweight preprocessing step, and no auxiliary language-specific methods such as part-of-speech tagging are required. No particular parameter tuning is needed in the original post-processing version; the thresholded version requires a parameter.

The two evaluation approaches emphasize different kinds of keyphrases: the distribution of evaluation keyphrases is concentrated on bi- and trigrams and even longer phrases in the scientific articles, whereas most of the Wikipedia links are uni- and bigrams. This has to be taken into account when selecting the evaluation approach for a keyphrase extraction system.

4.1.4 *Other tasks of IE and language-independent preprocessing*

Keyphrase extraction can be categorized as an *information extraction* (IE) task. A substantial part of the IE tasks collect *logical* or *structural* information from data (Yangarber, 2004), which often requires language-specific components. Also other tasks of IE exist in which language-independent methods can be used. In this section, only those tasks in the literature are discussed which have been approached with language-independent methods.

Syntactic analysis

Some NLP tasks aim to obtain syntactic features from text. The goal is to find syntactic patterns in text and use this information in further NLP analyses. Two of the most common tasks for syntactic analysis in which language-independent methods can be used are parsing and named entity recognition.

DETECTION OF SENTENCE BOUNDARIES A full stop character is a sign for a sentence boundary, but in addition, it codes, for example, abbreviations, initials and ordinal numbers. A language-independent and unsupervised method for detection of sentence boundaries was proposed by Kiss and Strunk (2006). The method does not use linguistic processing or orthographic information. The method first locates abbreviations as collocations, and other non-sentence-

boundary-full-stops, and then classifies all full stops with a likelihood ratio. The mean accuracy of sentence boundary detection on newspaper corpora in eleven languages was 98.74%. The languages were from the Germanic and Romance language groups, and Estonian and Turkish.

PARSING Parsing is the task of doing syntactic analysis for a sentence. Both rule-based and probabilistic parsers exist (Manning and Schütze, 1999, Ch. 10–12). Also unsupervised approaches have been proposed for part-of-speech tagging, but they usually need a dictionary of the possible tags for each word. Haghighi and Klein (2006) used a small list of labelled prototypes and no dictionary. Nivre *et al.* (2007) presented *MaltParser*, a data-driven dependency parser that uses a treebank in each language as the data for constructing the parser. The system has been used for a variety of languages: Bulgarian, Chinese, Czech, Danish, Dutch, English, German, Italian, Swedish, Turkish, representing languages in which the amount of morphology and flexibility of word order varies a lot. The parser achieved an unlabeled dependency accuracy above 80% for each language. Another approach is to induce just the syntactic clusters with unsupervised learning (Brown *et al.*, 1992; Clark, 2000). A language-independent approach for labelling the syntactic clusters was proposed by Schone and Jurafsky (2001), in which the only knowledge used was the information of language universals. Unfortunately, the study was conducted for the English language only.

MORPHEME SEGMENTATION Segmentation of morphemes in a word is an important preprocessing step especially for agglutinative languages, such as Finnish or Turkish. The morpheme segmentation task is to segment words into smaller meaningful units, morphemes. For example, 'flute'(*stem*) + 's'(*suffix*) and 'un'(*prefix*) + 'friend'(*stem*) + 'ly'(*suffix*) + 'ness'(*suffix*). Then words having the same base form can be located. Many unsupervised approaches have been proposed for morpheme segmentation. For example, there have been over 50 participating algorithms for the yearly organized Morpho Challenge competitions starting in 2005 (Kurimo *et al.*, 2010b). The Challenge was inspired by the development of a language-independent method *Morfessor* (Creutz and Lagus, 2007). Several languages have been tested in the Challenges: Finnish, Turkish, English, German and Arabic (Kurimo *et al.*, 2010b). Many of the participating methods participated in all the competition languages (Kurimo *et al.*, 2010a). One of the best-performing methods across languages used segment predictability and word segment alignment (Bernhard, 2008).

NAMED ENTITY RECOGNITION Named entity recognition (NER) aims to detect person and organization names and locations. There are many language-specific NER methods that use linguistic preprocessing and other resources. Two shared tasks for named entity recognition have been organized: for Spanish and Dutch in CoNLL-2002 (Tjong Kim Sang, 2002), and English and German in CoNLL-2003 (Tjong Kim Sang and De Meulder, 2003). Both the tasks claimed to be language-independent tasks but actually the participants used a lot of language-specific components. An exception is the system by Whitelaw and

Patrick (2003) who proposed a character-based probabilistic approach which uses only probabilistic features. However, the approach needs labelled training data for each language.

A step forward towards language independence was taken in an EM-style bootstrapping algorithm for named entity recognition (Cucerzan and Yarowsky, 1999). The algorithm needs a list of seed named entities, the size being the order of one hundred items for each named entity class. The method does not use linguistic preprocessing but uses capitalization information, word separators and a stop word list. The model has parameters for many language-specific features, such as the existence of word separators, the linguistic meaning of the first character of a word, etc. The analyzed languages were Romanian, English, Greek, Turkish and Hindi.

SPELLCHECKING A language-independent spellchecking system that does not use any manually annotated training data was proposed by Whitelaw *et al.* (2009). The system used data collected from the Web and processed it using two assumptions: misspellings tend to be orthographically similar to the correct word, and most of the words are spelled correctly. Mutually close words were collected from the Web using Levenshtein–Damerau edit distance and the most frequent word of each group was selected to be the 'correct' spelling. The 'misspelled' versions were further filtered in order to get a list in which the correct term was 10 times as frequent as the most common misspelling. The system parameters required supervised learning but the training data was produced using artificial misspellings for news texts which were assumed to be well-spelled.

Semantic analysis

Another set of information extraction methods consist of methods for semantic analysis of language. Language-independent approaches of some level have been proposed for almost any task of information extraction. However, some of the most popular tasks within semantic analysis seem to be too difficult to solve entirely with unsupervised and language-independent methods. One example is *question–answering*, for which subtasks involving clustering and ranking the answer set have been conducted in a language-independent manner, but analysis of the questions needs some annotated data (Whittaker *et al.*, 2006; Solorio *et al.*, 2004). Another task is *sentiment analysis* which seems to require labelled language-specific data. An important semantic task, *word sense disambiguation* (WSD), is also a task approached usually with supervised methods. WSD and the related tasks are further discussed in Section 4.3.2.

DETECTION OF SUBJECT BOUNDARIES The task to detect the switch from one topic to another in running text is called detection of subject boundaries. Richmond *et al.* (1997) found subject boundaries using a statistical approach: they calculated a significance value for each word, based on local burstiness and global frequency. Burstiness was defined as the distance to the nearest occurrence of the same word. As a preprocessing step of the method, different word forms were combined. The method used for combining was not reported

but statistical (and thus language-independent) approaches for stemming exist, for example *Morfessor* (Creutz and Lagus, 2007).

QUESTION CLASSIFICATION In open-domain question answering, a user asks a question and a system answers in a natural language. Question classification is part of the answering process and is needed to reduce the search space of the answers. Solorio *et al.* (2004) proposed a method that exploits lexical features and the Internet. The system used a stop word list and applied a set of heuristics to each language: A question 'Who is the President of the French Republic' was submitted to a search engine by adding an ending 'is a person', 'is a place', 'is a date', etc., and the number of hits was considered as an evidence of the semantic category of the word. First, all the content words were used ('President French Republic is a place', ...). If no matches, the last word was removed ('President French is a place', ...), and so on. The used classifier was a polynomial SVM. The experiments were run for English, Italian and Spanish and the best classification accuracies (with a representation incorporating also bag-of-words or 4 or 5 first letters of each word) for all the languages were between 81–89%.

Passage retrieval is used in the question–answering task to find text passages that contain relevant terms to the question. A language-independent method for passage retrieval was proposed by Gómez-Soriano *et al.* (2005) and applied to Spanish, French and Italian.

DOCUMENT SUMMARIZATION Document summarization is the task of reducing a text to the main points, using full sentences. Mihalcea and Tarau (2005) used graph-based, modified *PageRank* (Brin and Page, 1998) and *hyperlinked induced topic search* (HITS) (Kleinberg, 1999) algorithms for single-document and multi-document extractive summarization. They ranked the sentences in documents using vertices which were connected according to the content overlap in the sentences. The overlap was determined either as the number of common tokens or concentrating on a certain syntactic category. The experiments were run for English and Brazilian Portuguese. Another work for multi-document extractive summarization used *LexPageRank* (Erkan and Radev, 2004), also a version of *PageRank*. The experiments were run for English but there seems to be no language-specific components used. Also keyphrase extraction can be seen as creating very short summaries of documents, into single words or phrases.

OPINION DETECTION Sentiment analysis and opinion detection tasks have gained much interest, but not many language-independent approaches exist. Zagibalov and Carroll (2008) did opinion detection for English, simplified and traditional Chinese, and Japanese. Their topic relevance detection was unsupervised but opinion classification used a small number of manually selected words for each language, a set which was then automatically expanded.

4.2 TAXONOMY LEARNING

A taxonomy is a simplified version of an ontology, which was discussed in Section 2.2.2. A taxonomy is a structure of concepts, similarly to an ontology, but the relations between the concepts in a taxonomy are hierarchical parent–child relations. Furthermore, a taxonomy may not have linguistic labels for the concepts. Some examples of taxonomies are scientific classifications of animals and plants, family trees, and the items sold in an online store, organized hierarchically into groups and sub-groups. A language-independent approach for taxonomy learning was presented in Publication III. From now on, and as defined in the publication, the term *taxonomy* is used for a hierarchy of concepts, and the term *ontology* for a taxonomy that also has labels for the concepts, including the internal nodes.

Similarly to ontologies, taxonomies have traditionally been constructed manually. One of the first approaches for automatic extraction of taxonomic relations was proposed by Amsler (1981), who created a taxonomy of English nouns and verbs using dictionary definitions. Currently, the proposed approaches for taxonomy induction from text can be divided in four paradigms: the use of lexico-syntactic patterns, hierarchical clustering, document collections, and the use of meta information. Different approaches for taxonomy learning are discussed in more detail in Publication III. Here, only the approaches with some level of language independence are discussed.

Fallucchi and Zanzotto (2011) proposed a probabilistic method that exploits vector space model techniques (see, e.g., Salton *et al.*, 1975) in taxonomy learning. Other approaches using vector space models derived term hierarchies automatically from text using hierarchical clustering algorithms (Faure and Nédellec, 1998; Cimiano *et al.*, 2005; Grefenstette, 1994). The use of statistical methods in the extraction of taxonomic relations was discussed by Maedche *et al.* (2003), including hierarchical and non-hierarchical clustering, similarity measures and different linking schemes. Snow *et al.* (2006) proposed a clustering approach for taxonomy learning that incorporates evidence from multiple classifiers to optimize the entire structure of the taxonomy.

Taxonomy learning paradigms originating in the information retrieval community are based on the use of documents as descriptions of concepts. An example is the work by Sanderson and Croft (1999). Sánchez and Moreno (2005) presented an automatic and unsupervised methodology for creating a taxonomy for a certain domain. The system used a Web search engine for a seed word to collect documents from which the knowledge for building a taxonomy was extracted. Kozareva *et al.* (2009) proposed a supervised bootstrapping algorithm which created a taxonomy from Web documents, starting with two seed words. Velardi *et al.* (2007) presented a semi-automatic approach to extract domain-specific taxonomies from Web documents. The approach presented in Publication III is closest to these approaches.

The emergence of collaborative tagging systems and other social media has made possible the use of meta information, such as tags. An example is the approach to taxonomy learning from folksonomies (Benz and Hotho, 2007). Ponzetto and Strübe (2007) learned taxonomies from Wikipedia by using its cat-

egories as concepts in a semantic network, but relied on NP chunks and other language-specific components. Wong (2009) covered the literature of ontology and taxonomy learning comprehensively. The work also introduced the *tree-traversing ants* (TTA) clustering technique for learning taxonomic relations. The method uses Google search engine results and Wikipedia Categories information.

4.2.1 *Language-independent taxonomy learning (Publication III)*

In Publication III, a language-independent and automatic method to create a taxonomy was presented. Unlike the other taxonomy learning methods in the literature which were tested with one language only, the proposed method was tested with three languages: Finnish, English and Spanish. Another difference to many of the other methods is that Publication III proposes a statistical and unsupervised approach. The method treats a collection of encyclopedia entries or other topic-related documents as concept definitions, a view shared with Gabrilovich and Markovitch (2007). The proposed method uses the concept definitions as the basis for automatic taxonomy learning. The method involves feature extraction from the encyclopedia documents and hierarchical clustering of the feature vectors. One of the three proposed feature extraction approaches is a language-independent approach which combines statistical stemming with *Morfessor* (Creutz and Lagus, 2007) and *Likey* keyphrase extraction (Publication I). The other two approaches use language- and domain-specific components: a combination of rule-based stemming[3] and *tf-idf* weighting; and a combination of rule-based stemming and fuzzy logic-based feature weighting and selection. The proposed methodology can be applied to any domain where concept definitions are available. Access to online sources or other knowledge bases is not needed after collecting the document set.

Three versions of *Likey* were tested in feature extraction: *Likey*, $Likey_N$ and $Likey_W$, with the original post-processing (see Section 4.1.1). The fuzzy logic-based feature extraction methods were *fcc* and *efcc*. Further details about the fuzzy logic system can be found in Publication III. The baseline method *tf-idf* was used as in Table 6.

METHOD To make the taxonomy construction possible, the following assumption was made: each concept is supposed to have a hypernym, i.e., a parent concept or node. The *self-organizing map* (SOM) (Kohonen, 2001) was used for creating an ordered space of the concept vectors on one level of hierarchy. The taxonomy creation was started from the top (zero-level) of the hierarchy and continued until the bottom (K-level). All the feature vectors in the taxonomy constituted the zero-level of the taxonomy. To obtain the first level, the document feature vectors were clustered. For example, a data set of cities in the world would have the *world* (the location of all the cities) as the root node

3 English: Porter stemmer `http://www.tartarus.org/~martin/PorterStemmer`,
Finnish: Snowball `http://snowball.tartarus.org/algorithms/finnish/stemmer.html`,
Spanish: Snowball `http://snowball.tartarus.org/algorithms/spanish/stemmer.html`

on the zero level in the taxonomy of cities. The cities that locate on the same *continent* would form a cluster on the first level.

Since several adjacent SOM units may constitute a cluster, hierarchical clustering was applied on top of the SOM lattice to obtain the cluster borders. Agglomerative hierarchical binary clustering was used for producing up to k clusters of the SOM units. Each cluster on the first level was further clustered with SOM and hierarchical clustering to obtain the second level of the taxonomy. In the city data example, the continents would be clustered separately to get clusters marking *countries* in each continent. This clustering procedure was continued until the required number of levels in the taxonomy was reached or if each cluster contained only one data item.

The approach presented in Publication III is related to the *tree-structured self-organizing map* (Koikkalainen and Oja, 1990), in which every SOM unit on the zero level has children on the next level, and further the *growing hierarchical self-organizing map* (Dittenbach *et al.*, 2000), in which the SOM units are expanded if a certain criterion is reached. Both the approaches expand single units instead of clusters as in our approach. We selected a SOM-based approach because SOM handles easily large-dimensional textual data (Kaski *et al.*, 1998), but of course any non-hierarchical clustering method would go here. Another possibility would have been to use hierarchical clustering directly but it would have needed an additional method to reduce the resulting (binary) tree into a small number of hierarchy levels.

EVALUATION METHOD As taxonomies are difficult and laborious to evaluate manually, an automatic evaluation approach was proposed in Publication III: the concepts of the learned taxonomy are labelled using evaluation data to be able to compare the taxonomy to a reference ontology. The evaluation measures were the global taxonomic precision *TP*, recall *TR*, and F-measure *TF* (Dellschaft and Staab, 2006).

For evaluation, the labels for the lowest-level (K-level) of the learned taxonomy were obtained from the titles of the Wikipedia articles. The parents of each concept were collected from the reference ontology. The majority of parent concepts in a cluster was chosen to be the label of the cluster, i.e., the hypernym of each concept in the cluster. In the case of two parent candidates of equal sizes the hypernym was selected randomly.

The selected evaluation measure allows each label only once in the hierarchy, that is, duplicate labels are forbidden. Therefore, for labelling purposes, clusters having both the same label and the same parent node were merged. This corresponds to a situation where the clustering has made a too fine-grained division between data items, which was not penalized in the evaluation. In contrast, in a case where the clusters with the same label had different parent nodes, the result was penalized by setting the cluster that had less concepts as unclassified. In this way, the double-labelling was penalized if the clusters located in different branches of the hierarchy but too fine-grained clustering was not penalized.

RESULTS The experiments were carried out for three European languages: English, Finnish, and Spanish. By using these languages from different language groups and families, it was shown that the methodology is applicable to a large set of languages. The data consisted of 166 Wikipedia articles about animals, the same animals for each language. The reference ontology for evaluation was manually collected from the Wikipedia articles: an animal hierarchy of three levels. An example clustering with $Likey_W$ for English is shown in Figure 5. The figure shows that the first-level clustering was not successfull in locating all the four Classes but found only two: *Mammalia* and *Aves*. The second clustering within the Class *Aves* was able to find three out of four Orders, and placed about half the animals into the correct clusters. Within the second Class, *Mammalia*, three out of five Orders were found and two extra non-mammal Orders, as well as a group of unclassified animals, were suggested. The recall of the Order groups within Class *Mammalia* is quite high, but precision is not, basically as a consequence of the missing Classes on the first level of the hierarchy.

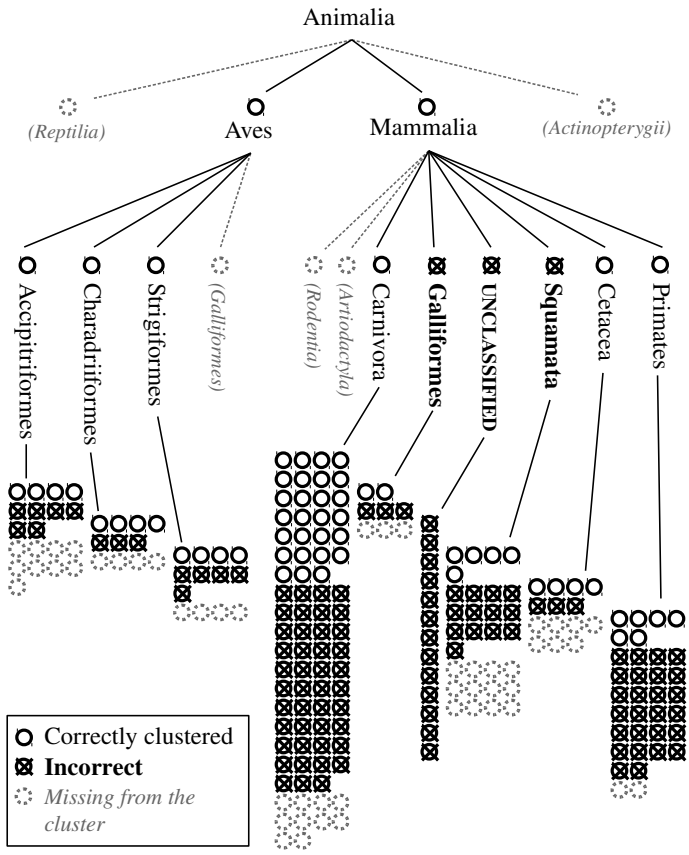


Figure 5: An example clustering of animals into a three-level taxonomy using $Likey_W$ for the English data set. Correct, incorrect and missing children are indicated for each parent node.

The results measured with the global taxonomic precision *TP*, recall *TR*, and F-measure *TF* measures for each language and feature extraction method are shown in Figure 6. Weighted $Likey_W$ got better recall but worse precision than *Likey* and $Likey_N$. The values of the *TF* scores were around 0.7 and thus it can be said that the language-independent methodology is able to achieve reasonably good results in solving the taxonomy learning problem. The relatively good results of *efcc* are understandable since it exploits extra information, the HTML page structure. The purely statistical *Likey* methods performed usually slightly poorer. Even though the baseline *tf-idf* is also a statistical method, it was combined with rule-based stemming, which seems to help for the Finnish language. Overall, the different results between the languages may stem from the fact that Wikipedia does not contain exactly the same information in different languages.

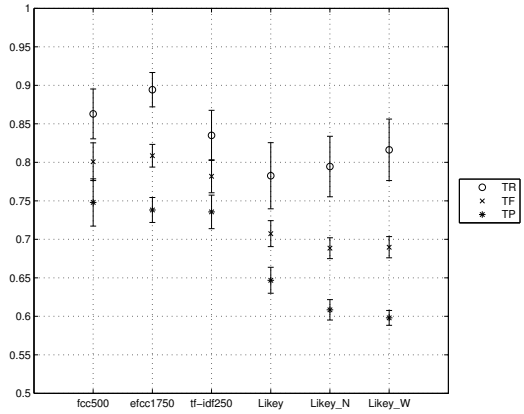
4.3 LEXICAL CHOICE AND DISAMBIGUATION

Ambiguity of words and redundancy are important features of language. However, these are the main problems in automatic understanding of natural languages. In this section, a study on near-synonym lexical choice (Publication IV) is discussed, as well as language-independent approaches for word sense disambiguation in the literature.

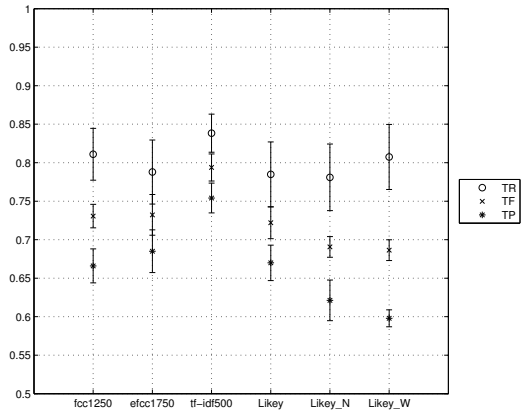
4.3.1 *Near-synonym lexical choice (Publication IV)*

Lexical choice is an important subtask in systems that generate natural language, such as machine translation, question–answering and summarization. The goal of *lexical choice* is to find a correct word to fill a lexical gap in a sentence, depending on the context. In the task of *near-synonym lexical choice*, the best alternative is selected out of a set of near-synonyms, which is a difficult problem because of the fine-grained differences between the meanings of the words. Some methods have been proposed for the problem in the literature (Edmonds, 1997; Wang and Hirst, 2010).

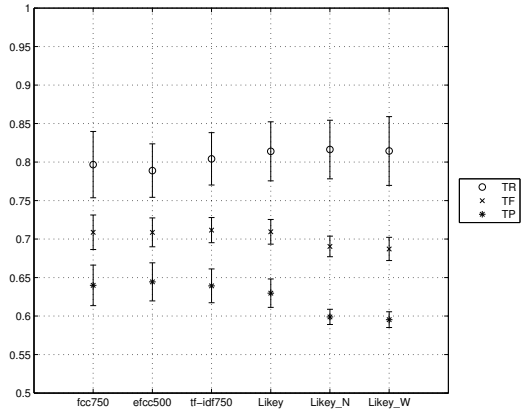
In Publication IV, different machine learning paradigms were studied for clustering word contexts in the near-synonym lexical choice task. In contrast to the language-independent approaches presented in the other publications of the thesis, this approach used an extensive set of over 650 linguistic features to represent the context of a word. One of the goals was to experiment how good accuracy can be obtained with different machine learning paradigms (unsupervised, semi-supervised and supervised), provided with an extensive set of language-specific features. However, the language specificity of the approach is not as deep-rooted as it may first look: the objective was to take *whatever linguistic features available* and apply machine learning to them, without the use of any language-specific rules or other knowledge. This kind of approach is general and could be applied to texts in any language for which some sort of linguistic analysis is available.



(a) English



(b) Finnish



(c) Spanish

Figure 6: Mean global taxonomic precision *TP*, recall *TR* and F-measure *TF* with standard deviations of the taxonomy learning results for (a) English, (b) Finnish, and (c) Spanish. The dimensions that give the best results are shown for *efcc*, *fcc*, and *tf-idf*. From Publication III.

DATA AND FEATURE SETS The data used in Publication IV is the *amph* data set (Arppe, 2008). It consists of 3404 occurrences of four *think* lexemes in Finnish with over 650 morphological, semantic, syntactic, and extra-linguistic features. The lexemes are 'ajatella', 'harkita', 'miettiä', 'pohtia', roughly corresponding to the English 'think', 'consider', 'reflect', and 'ponder', respectively. The data had been earlier collected from newsgroup postings and newspaper articles (Arppe, 2008) and is publicly available[4]. The distribution of the four lexemes in the data is given in Table 15.

Table 15: *Think* lexemes and their frequencies and percentages in the *amph* data set.

Lexeme	**Frequency**	**%**
1. 'ajatella' ('think')	1492	43.8
2. 'harkita' ('consider')	387	11.4
3. 'miettiä' ('reflect')	812	23.9
4. 'pohtia' ('ponder')	713	20.9
Total	3404	100.0

The *amph* data set had been morphologically and syntactically analysed with a computational implementation of functional dependency grammar for Finnish (Tapanainen and Järvinen, 1997), with manual validation and correction (Arppe, 2008). In addition, the analysis had been supplemented with semantic and structural subclassifications of syntactic arguments and the verb-chain. The data set consists of 216 binary atomic features and 435 binary feature combinations. The sizes of the feature sets and some examples falling to each category are shown in Table 16.

To give an example of the features, a sentence in the data set 'Hän ei aina harkinnut sanojaan' ('He did not always consider his words') was analyzed according to the features. A sample of the features is shown in Table 17.

In Publication IV, two original feature sets were used: FULL, all 651 features, and ATOMIC, atomic features only (216 features). Their performances were compared to a feature set M6, which had been manually selected from the FULL feature set by an expert (Arppe, 2008, p. 194). In addition, one more feature set was selected automatically in order to get a better feature set for the classification task. The *forward feature selection* method was applied using the supervised kNN classifier with $k = \{1,3,5,10\}$ as the evaluation criteria to select those features that best distinguish between the lexemes. The features were added incrementally from the FULL feature set. The results of the forward feature selection can be seen in Figure 7. The 5NN method was the quickest to reach the highest accuracy level of around 0.65–0.66 at about 40 features and thus it was used as an automatically selected feature set FS40 in the classification experiments.

All the experiments were run with 20-fold cross-validation. The statistical significances were calculated with the *1-sided Wilcoxon signed rank* test on the significance level of 0.05. The selected baseline method classified all data items to the largest category, lexeme 1, resulting in an accuracy of 0.44.

4 http://www.csc.fi/english/research/software/amph

Table 16: Atomic (A) and combined (C) features of the four *think* lexemes in the *amph* data set.

Feature group		Freq.
A1. Morphology	The lexeme or other verbs in the verb chain in conditional mood, passive voice, …	75
A2. Syntactic arg's of the lexeme	Functional role, e.g., agent, patient, temporal, …	22
A3. A word in any syntactic position	Whether e.g., 'always', 'no', … are syntactically related to the lexeme	68
A4. Extra-linguistic	Information about the source, author, …	51
		216
C1. Syntactic & semantic	Syntax & semantic class of the argument, e.g., human individual, …	173
C2. Syntactic & phrase-structure	Direct quote, indirect question, 'that' clause as patient	13
C3. Syntactic arg. & base-form lexemes	E.g., 'he'/'she' as agent/patient, 'always' as temporal arg., 'carefully' as manner arg.	63
C4. Syntactic & morphology	Morphological analysis for argument lexemes	186
		435

Table 17: Example features for sentence 'Hän ei aina harkinnut sanojaan' ('He did not always consider his words'). The verb chain of the sentence is 'ei harkinnut' ('did not consider'). The features are shown in italics.

Group	Examples
A1.	The verb chain is *third person singular, past tense, active voice,* …
A2.	The verb chain contains *negative auxiliary verb* ('ei' ('did not')); has *temporal argument* ('aina' ('always')), …
A3.	The words *'aina' ('always'), 'ei' ('not'), 'hän' ('he')* are in syntactic relationship with the lexeme.
A4.	The sentence is from *newspaper*; 'harkita' ('consider') is the *first* think *lexeme* in the document, …
C1.	The *agent* of the sentence ('hän' ('he')) is *an individual,* …
C3.	*'Ei' ('did not') is negative auxiliary,* …
C4.	*The negative auxiliary* 'ei' ('did not') is *third person singular,* …

RESULTS Two unsupervised methods applied to the data, the *self-organizing map* (SOM) (Kohonen, 2001) and *independent component analysis* (ICA) (Comon, 1994), did not succeed in extracting components that match well with the *think* lexemes in the classification task. The resulting components clearly represented some underlying structure in the data set, but different from the

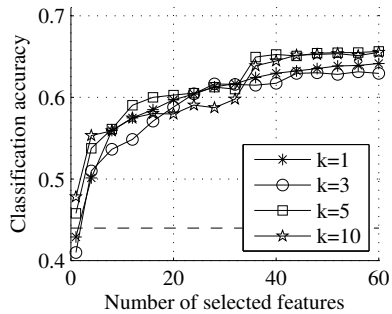


Figure 7: Supervised classification accuracy of kNN, $k = \{1,3,5,10\}$, for feature selection. The dashed horizontal line shows classification accuracy of a majority class baseline classifier. From Publication IV.

desired classification, which shows that the data contain similarity structure stronger than the division between the lexeme classes.

Another unsupervised method, *k-means*, was applied to the four feature sets, resulting in classification accuracies shown in Table 18. As can be seen in the table, the larger the number of clusters, the more coherent the clusters and the better the accuracies which approach 100%. For all numbers of clusters k, the automatically selected feature set FS40 performed significantly better than any of the other tested feature sets, even though it contains the smallest number of features. This naturally is to be expected, since FS40 was chosen to maximize the label accuracy. Nevertheless, clustering into four categories did not differ much from the baseline accuracy of 0.44.

Table 18: Unsupervised classification accuracy of *k-means* using the four feature sets. FS40 performs significantly better for all numbers of clusters k (in bold) against all other feature sets.

	FULL	**ATOMIC**	**FS40**	**M6** (Arppe, 2008)
k	**Avg**	**Avg**	**Avg**	**Avg**
4	0.44	0.44	**0.45**	0.44
6	0.44	0.44	**0.47**	0.45
8	0.44	0.44	**0.50**	0.46
10	0.44	0.45	**0.51**	0.47
20	0.46	0.48	**0.55**	0.49
30	0.49	0.48	**0.56**	0.50
50	0.52	0.50	**0.57**	0.54
100	0.54	0.51	**0.59**	0.56

As a semi-supervised method, a *semi-supervised* kNN with $k = \{1,3,5,10\}$ was used. For $k > 1$, a straightforward extension from the 1NN classifier by Zhu and Goldberg (2009) was implemented. The experiments were run by using labelled data of 5–100% of all training data. The average classification accuracies with

the ATOMIC feature set are shown in Figure 8. When at least 15% of the data was labelled, *semi-supervised* kNN with all the tested values of k performed better than the baseline. 10NN reached the highest accuracy. Statistically significant differences were found between 1NN and the other methods when at least 50% of the data was labelled. Similar results were obtained also with the other feature sets.

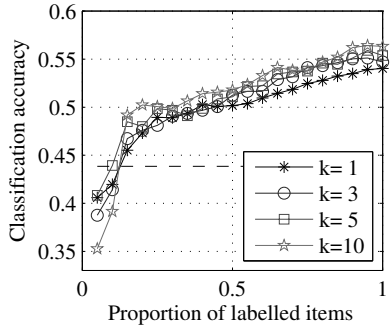


Figure 8: Semi-supervised classification accuracy of *semi-supervised* kNN using ATOMIC feature set, varying the proportion of labelled data items between 0.05–1. The dashed line shows the baseline. From Publication IV.

Unsupervised and semi-supervised methods were not able to find the structures that differentiate the four lexemes very well. Thus, some supervised methods were also tested: *feed-forward artificial neural network* (ANN) (see, e.g., Haykin, 1994) with a hidden layer of 20 neurons, *multinomial logistic regression* (MNR) (McCullagh and Nelder, 1990), and *k nearest neighbours* (kNN) (Cover and Hart, 1967) with a varying number of neighbours. Table 19 shows the classification accuracies of the supervised methods. The supervised results of FULL and FS40 are significantly better than Arppe's manually selected M6 with the ANN classifier. The FULL set obtained the significantly best results with MNR. For kNN, FULL and FS40 performed better than the Arppe's M6, the FS40 obtaining usually significantly the best accuracies. The results showed that FS40, constructed with supervised feature selection, had substantially smaller dimensionality but, at the same time, offered significant increase in accuracy with supervised kNN, compared to the FULL feature set.

DISCUSSION The best classification accuracy obtained in the task was 0.66 with MNR for the FULL feature set with 651 features. The automatically selected feature set FS40 of only 40 features performed very well overall: even though losing to the other feature sets with MNR, it achieved similar or higher accuracy than any of the other feature sets with with ANN, kNN and *k-means*.

The FS40 set consisted mostly of syntactic features (29 features, including 12 semantic subtypes), together with a few morphological and extra-linguistic features. Nine out of the first ten selected features were syntactic and two out of the first seven features were combinations of syntactic and semantic features. The first features in FS40 also correspond to the manually selected feature set M6: six out of the first eight automatically selected features can be found in

Table 19: Supervised classification accuracies of ANN, MNR, and kNN with different number of neighbours k using the four feature sets. The result for the significantly best feature set is printed in bold for each method (row). For kNN, the best values of k for each feature set are underlined.

		FULL	ATOMIC	FS40	M6 (Arppe, 2008)
		Avg	Avg	Avg	Avg
ANN		**0.62**	0.59	**0.64**	0.59
MNR		**0.66**[1]	0.61[1]	0.60	0.63
kNN	$k=1$	**0.60**	0.54	0.47	0.53
	3	0.60	0.55	**0.64**	0.58
	5	0.60	0.56	**0.65**	0.58
	10	0.61	0.57	**0.63**	0.59
	20	0.60	0.56	**0.64**	0.59
	30	0.59	0.56	**0.63**	0.58
	50	0.57	0.54	**0.62**	0.57
	100	0.54	0.54	**0.61**	0.56

[1] Computed for the first 150 principal components.

the M6 set, but the remainders of the feature sets are almost distinct. Because the automatically selected small feature set performed comparatively well in the near-synonym lexical choice task and it contained features from every category shown in Table 16, it could be concluded that besides syntactic features, also semantic and morphological features are required in near-synonym lexical choice. It is natural to think that linguistic analysis of the word context would yield almost perfect performance in the task but we showed that too many features, even linguistic, introduce noise and make lexical choice a difficult problem for unsupervised and semi-supervised methods. Further, the contexts of the lexemes are not completely distinct and in some cases more than one lexeme would be a perfect choice for a context.

4.3.2 *Word sense disambiguation*

Word sense disambiguation (WSD), which was already mentioned in Section 2.2, is a very important NLP problem due to the ambiguity of natural language. The goal of WSD is to find the correct sense of an ambiguous word, using the word context as evidence. WSD is typically a supervised task and thus cannot be language-independent. For a textbook description of WSD, see, e.g., Manning and Schütze (1999, Ch. 7) or Jurafsky and Martin (2009, Ch. 20). Lexical choice (discussed in the previous section) and WSD can both use the same categorization and clustering methods, even though they are distinct tasks. In WSD, the surface forms of concepts are the same and the different meanings have to be disambiguated. In lexical choice, there are different surface forms with closely related meanings.

There are also other related tasks in the NLP literature. *Word sense induction* (WSI) or *word sense discrimination* is an unsupervised variant of WSD. In WSI,

different uses of a word type are separated (or clustered) from each other without prior knowledge about the number of alternative senses or the nature of the difference between the senses. An unsupervised word sense discrimination approach for English was proposed by Purandare (2003) and Purandare and Pedersen (2004). It was further used for *name discrimination* for English (Pedersen *et al.*, 2005) and additionally for Bulgarian, Romanian, and Spanish (Pedersen *et al.*, 2006). The methods use log-likelihood ratios of bigrams. In (Pedersen *et al.*, 2006), the word context was represented with a second-order context vector.

Lexical substitution is a task in which a word in a context is to be replaced with a synonymous word that is also suitable for the context (McCarthy, 2002). However, there is not a predefined list of possible answers available as in WSD or lexical choice. Lexical substitution has gained some popularity in the SemEval tasks (McCarthy and Navigli, 2007; Mihalcea *et al.*, 2010).

In the information retrieval community, lexical choice is known as *query expansion* (Voorhees, 1994). Machine translation (MT) is also a large application area (Apidianaki, 2009; Carpuat and Wu, 2007; Bangalore *et al.*, 2007). In MT, the task is often referred to as *lexical selection*, where the target word is selected from a set of possible translations. Many vector space models have been evaluated in lexical choice tasks, such as the synonym part of the TOEFL language test (Landauer and Dumais, 1997; Rapp, 2002; Sahlgren, 2006b).

4.4 SEMANTIC DOCUMENT REPRESENTATION

When transferring the semantic contents of documents into a form understandable by computers, vector spaces are a common choice. As discussed earlier in Section 3.3, vector spaces are constructed using feature selection or feature extraction methods. Some of the most common tasks using vector space models are information retrieval (Salton *et al.*, 1975) and document clustering in which semantically similar documents are grouped together. In this section, two evaluation frameworks for document representation are considered. First, different dimensionality reduction methods are tested in a *k-means* document clustering application. The setting is language-independent and fully automatic, even though the number of clusters was selected manually and a rule-based stemmer was applied in order to get results comparable to other studies in the literature. Second, a language-independent and fully automatic direct evaluation method for document representation is proposed.

4.4.1 *Dimensionality reduction and distance measures (Publication V)*

The dimensionalities of document collections are often very large, for example thousands or tens of thousands of words. In the document clustering task, it is common to reduce the original dimensionality for computational reasons. For measuring the distances between documents, cosine distance is widely seen as the best choice (Sahlgren, 2001; Bullinaria and Levy, 2007). In Publication V, *k-means* document clustering results were compared with three dimensionality reduction methods and a selection of distance measures: Euclidean,

standardized Euclidean, city block, Chebychev, cosine, correlation, Spearman, Bray-Curtis, a modified Bray-Curtis, and Canberra (shown in Table 7). Some of the distance measures had been tested earlier for document clustering (Huang, 2008; Madylova and Öğüdücü, 2009; Schenker *et al.*, 2003; Strehl *et al.*, 2000). The dimensionality reduction methods tested included PCA, SVD, and a naive approach that selects terms that have the largest *tf-idf* weight in the document set. The goal of the study was to show how the distance measures perform with normalization, different dimensionality reduction methods and a range of target dimensionalities.

DATA AND EVALUATION The goal was to cluster topically similar documents together. The clustering performance was measured using evaluation documents that were labelled with topic categories. The experiments were run with three standard datasets in English: NEWSGROUP and REUTERS[5], and CLASSIC[6]. In addition, a HINDI data set was collected from an online news service[7]. The numbers of evaluation categories in the data sets were 20, 8, 4, 4, respectively. The term–document matrices were created by applying standard preprocessing: lowercasing, punctuation and stop word removal, and stemming. Low-frequency words were removed and the rest of the words weighted with the *tf-idf* weighting scheme. Both unnormalized and l_2-normalized weights were used in the experiments. The evaluation measure was the *normalized mutual information* (NMI) score.

RESULTS Cosine, correlation and Spearman measures performed best in the classification task for all data sets and all target dimensionalities. With l_2-normalization, also Euclidean gave good results. In Figure 9, the effect of l_2-normalization is shown with PCA dimensionality reduction on the REUTERS data and with SVD on the NEWSGROUP data. The NMI score was used for comparing the obtained clusters to the document categories. The dimensionality was reduced into target dimensionalities ranging between 2 and 1000 and the clustering was run with *k-means* using all the distance measures.

Normalization affected neither cosine nor correlation measures in large dimensionalities but decreased the performance below dimensionality 10 with PCA dimensionality reduction. On the contrary, the other dimensionality reduction methods performed better with normalized data for almost all dimensionalities. Especially Bray-Curtis, both the Euclidean measures, and Spearman benefited from the normalization. An exception was Canberra which reached lower scores with normalized than unnormalized data above dimensionality 100.

The mean NMI results after SVD, PCA and *tf-idf* dimensionality reduction are shown in Figure 10 for all the four data sets: CLASSIC, REUTERS, NEWSGROUP, and HINDI, with l_2-normalization. The overall best results did not increase compared to the original dimensionalities, except for the small HINDI data set, which seems to require some smoothing. For HINDI, the best NMI score in the

5 http://web.ist.utl.pt/~acardoso/datasets/
6 ftp://ftp.cs.cornell.edu/pub/smart/
7 http://www.24dunia.com/hindi.html

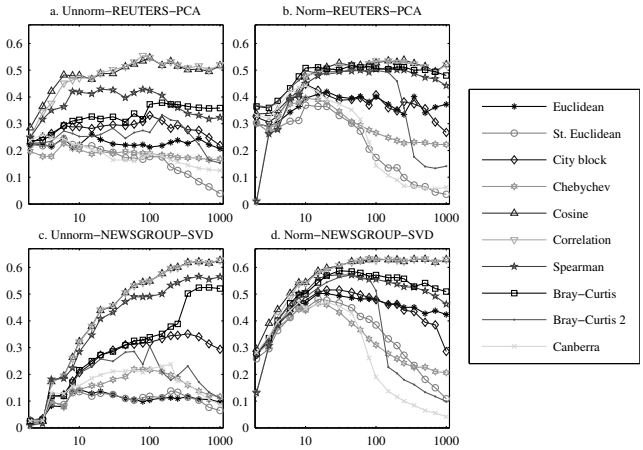


Figure 9: Mean NMI of unnormalized and l_2-normalized data for target dimensions 2–1000. From Publication V.

original dimensionality was 0.434 with the correlation distance measure. The score with cosine rises over 0.5 in the reduced dimensionalities.

As was also suggested in the literature, cosine and correlation distances performed best almost always in target dimensionalities above 10. However, for target dimension 10 or below, all the distance measures gave very similar results and the dimensionality reduction method had the major role in the performance. Thus for small target dimensionalities, the choice of the distance measure is not very important. Further, for the small dimensionalities, PCA performed better than SVD.

For target dimensionalities above 10 the differences between SVD and PCA were very small, except for Spearman which worked better with PCA than SVD. Spearman performed even better with *tf-idf* dimensionality reduction. The performance of the less-known Bray-Curtis was very stationary and usually within the three best measures throughout the target dimensionalities with both SVD and PCA, which suggests that it would be a good choice in a general task with several target dimensionalities. Standardized Euclidean worked with *tf-idf* dimensionality reduction, but gave poor results with the other dimensionality reduction methods. Canberra, Bray-Curtis 2, and Chebychev were always among the worst-performing measures. Standardized Euclidean, city block, Chebychev and Bray-Curtis did very poor performance in the original dimensionality but benefited from the SVD and PCA dimensionality reductions.

Dimensionality reduction is a standard way to decrease the computational costs of processing high-dimensional document matrices. Dimensionality reduction seems not to increase the performance of the best measures but indeed helps poorer performing methods which achieve almost the same performance level with the best ones with small target dimensionalities $D < 20$.

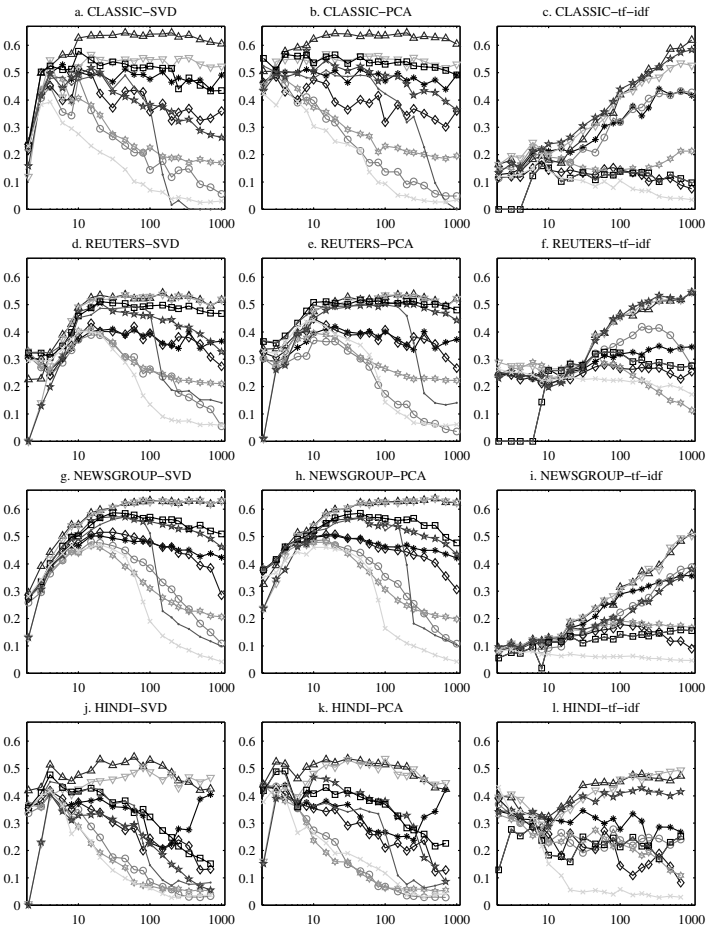


Figure 10: Mean NMI after dimensionality reduction into dimensions 2–1000 for l_2-normalized data. See Figure 9 for the legend. From Publication V.

4.4.2 *Document representation evaluation (Publication VI)*

The evaluation methods for document vector representation can be categorized as direct and indirect methods (Sahlgren, 2006a). The *direct* methods evaluate the vector space structure using external data, whereas the *indirect* methods evaluate the vector space performance in an application. The previous study in Publication V applied indirect evaluation in document clustering. Information retrieval (IR) research (see, e.g., Manning *et al.*, 2008) commonly use the indirect evaluation of a vector space: the quality of a vector representation is typically measured using IR results for evaluation. In document retrieval, for example, the evaluation is based on measuring how well the IR system is able to rank documents according to the query. Other tasks for indirect evaluation include word sense induction (Schütze, 1992), lexicon extraction from comparable corpora (Rapp, 1995; Gaussier *et al.*, 2004), and comparison to bilingual

lexica originally intended for human use (Sahlgren and Karlgren, 2005). In an approach by Besançon and Rajman (2002), documents from a bilingual corpus were mapped onto two separate monolingual vector spaces and the indirect evaluation was conducted by comparing the nearest neighbours of the documents.

In direct evaluation of document representations, the proposed methods find semantic relations within the vector space, such as synonyms, antonyms, sub- or superconcepts. As the external evaluation data, many corpora intended originally for human use have been applied, such as lexica, priming data, association norms, or synonym and antonym tests (Sahlgren, 2006a). Examples of the data sets are the Test for English as a Foreign Language (TOEFL), first proposed by Landauer and Dumais (1997), the Test of English as a Second Language (ESL) multiple-choice synonym questions (Turney, 2001), and the SAT college entrance exam (Turney, 2005). Other evaluation data are for example word associations (Kiss *et al.*, 1973; Nelson *et al.*, 1998) and thesauri (Roget, 1911; Bernard, 1990). Also more structured lexical databases are available, such as *WordNet* (Miller, 1995; Fellbaum, 1998) and other ontologies of different areas and languages. Human evaluators have also been used (e.g., Mitchell and Lapata, 2008; Zesch and Gurevych, 2009). Also psychological evidence, such as reaction times (Virpioja *et al.*, 2011a) and eye movements (Salojärvi *et al.*, 2004) have been used in direct evaluation.

The evaluation approaches in the literature have their disadvantages: Indirect evaluation in an application is often time-consuming and the results may not generalize to other applications, whereas direct evaluations which measure the amount of captured semantic information require usually human evaluators or annotated data sets. A novel direct evaluation method for document representations was proposed in Publication VI. The method uses unsupervised learning and is language- and domain-independent.

METHOD The proposed evaluation method is based on the assumption that feature extraction retains as much as possible of the semantic content of a text. A bilingual corpus contains the same semantic content on both parts of the corpus. The evaluation is performed for a bilingual corpus by applying *canonical correlation analysis* (CCA) (Hotelling, 1936), which can be used for finding linear relationship between two data sets. The proposed method framework is presented in Figure 11. The corresponding documents s and t in two languages are generated from the same semantic vector $\mathbf{z}$ in a language-independent semantic space, through language-specific semantic subspaces $\mathbf{z}_s$ and $\mathbf{z}_t$, respectively. A feature extraction process (which is to be evaluated) transforms aligned document collections $\mathbf{S}$ and $\mathbf{T}$ to feature matrices $\mathbf{X}$ and $\mathbf{Y}$. $\mathbf{X}$ and $\mathbf{Y}$ are projected onto a common vector space using CCA. The proposed evaluation method compares the projections $\mathbf{U}$ and $\mathbf{V}$, respectively, in the common vector space and evaluates whether the semantic contents has been retained in the feature extraction process. More details about the model can be found in Publication VI.

RESULTS As evaluation measures, the *sum of correlations* and *Gaussian mutual information* were introduced. The latter is the mutual information with

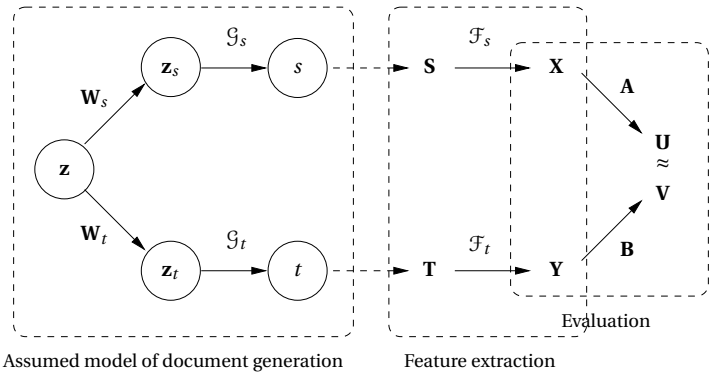


Figure 11: The evaluation process using the corresponding documents s and t in a bilingual corpus. From Publication VI.

the assumption that the features are Gaussian, which assumption, however, does not actually hold. Validation of the proposed evaluation method was conducted by comparing the results with known facts in the literature related to the best dimensionality reduction methods and weighting for language data, the amount of data, phrases as features compared to single words as features, and similarities between languages. Also validation in a sentence matching task and manual validation of translation pairs calculated with canonical factor loadings (Harman, 1960; Rummel, 1970) was conducted.

The evaluation results were intuitive and agreed with the previous findings in the literature, the validation facts. Two of the five validation experiments are summarized here. The first one is based on the known fact that due to morphological and syntactic similarities, closely related languages are supposed to correlate better than distant languages (Besançon and Rajman, 2002; Chew and Abdelali, 2007; Sadeniemi *et al.*, 2008). The validation experiment used the same feature extraction method for different language pairs and compared the correlation sum results between the language pairs. The results are shown in Figure 12: Danish and Swedish, two closely related Northern Germanic languages, are the most correlated within a set including Western Germanic languages English and German, and Finno-Ugric Finnish.

The second summarized validation experiment in Publication VI was to compare the evaluation results with manual validation of word translations. In the experiment, a range of feature extraction approaches were used: three settings of the SVD dimensionality reduction method, the most frequent words (W-FreqSet), inner product of the longest sentences (S-LenSet), and projecting to nearly orthogonal vectors (W-RandProj). More details about the tested feature extraction approaches can be found in Publication VI. For the resulting feature vectors, canonical factor loadings were calculated to find for both languages which original words contributed in each canonical variate. The contributing words were supposed to be translation pairs between the two languages. The results of the manual validation between Finnish and English are shown in Table 20. As a conclusion of the manual validation of the evaluation measure, the

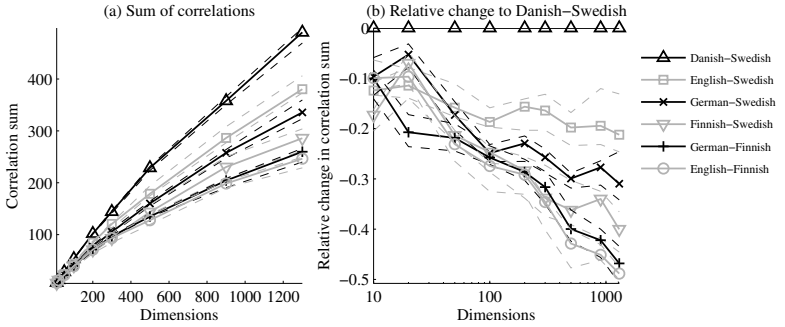


Figure 12: Correlation sums for test data with different language pairs: (a) absolute sum of correlations and (b) relative change compared to Danish–Swedish. From Publication VI.

number of correct translations found in the corresponding variables gave very similar results to the correlation sum.

Table 20: The results of the manual validation: the correlation sum is compared to the number of correct translations, using a number of document representation settings.

Method	**Correlation sum (rank)**	**Correct translations (rank)**
Weighted SVD, year data	41.5 (1.)	810 (1.)
Weighted SVD	38.7 (2.)	787 (2.)
SVD	29.9 (3.)	642 (4.)
W-FreqSet	28.7 (4.)	773 (3.)
S-LenSet	24.3 (5.)	516 (5.)
W-RandProj	12.0 (6.)	421 (6.)

The proposed direct evaluation method for document representations can be used for evaluating document representations in various NLP tasks, such as information retrieval or word sense disambiguation. The method requires only an additional bilingual corpus and running the feature extraction for both languages. The choice of the aligned bilingual document collection may affect the evaluation results: closely related languages may easily involve also other correlated features than the semantic ones. Thus for obtaining semantic features only, choosing languages from different language families would be recommended.

4.5 USER MODELLING AND SUBJECTIVITY

Every individual has an own unique collection of background knowledge, vocabulary and preferences. However, most of the NLP applications assume a general objective user with a single understanding of the world. They further assume that every speaker of a language use exactly the same language. In

other areas of research, the Internet and the development of software technology has made it possible to collect information about the variations between users through online behaviour or from other electronic sources. *User modelling* aims to get information about the user and adapt the system according to the user. One issue is how to adapt the user models over time (Ardissono *et al.*, 2001). In the area of NLP, user modelling is closely related to the concept of *subjectivity* in language use, discussed earlier in Section 2.1.3: the aim is to find the subjective conceptualizations and preferences in language use.

In Web-related tasks such as information retrieval, statistical methods and, in particular, machine learning algorithms have been proposed for general user modelling purposes (Billsus and Pazzani, 2000). A traditional approach to model users is that the users manually provide the required user profiling information through Web forms or questionnaires (e.g., Ardissono *et al.*, 2001). More recent approaches include, for example, relevance feedback given by the users (Wærn, 2004) or tagging (folksonomies) (Carmagnola *et al.*, 2008).

Automatic user modelling approaches that do not need any extra actions from the user are concentrated on the Web applications. Perugini (2010) personalised the contents of a Web page according to the user's interaction with the site by clicking the hyperlinks. Another approach is to create user profiles from the information the user had read earlier (Nanas *et al.*, 2010). In this study, content-based filtering was applied to selecting the most suitable news items and scientific articles for a user. Teevan *et al.* (2005) proposed a method for personalised information retrieval by re-ranking search engine results according to a user model. They adapted the results by weighting (expanded) query terms with their existence in user's personal content, such as visited Web pages, e-mail messages, calendar items and stored documents on a client machine. Some further user modelling examples are a recommender system of Web content (Adomavicius and Tuzhilin, 2005), a personalised search engine (Magnini and Strapparava, 2001) and question–answering (Quarteroni and Manandhar, 2006, 2007).

4.5.1 *User-specific difficulty of text documents (Publication VII)*

In this thesis, user modelling has been applied to a novel approach for text document difficulty assessment (Publication VII): the goal is find out how a document corresponds to the expertise level of a user. The proposed method could be used for example within a customized search system to select the search results according to the user's expertise level on different domains. Texts are written for different uses and thus for people having different levels of expertise on the domain. Texts intended for professionals in a certain domain may not be understandable at all by a lay person, and texts for lay people may not contain all the detailed information needed by a professional. The difficulty of understanding text was discussed earlier in Section 2.1.3.

A number of techniques have been proposed in the literature for the assessment of the difficulty level of text, ranging from manually calculated readability formulas, e.g., *automated readability index* (ARI) (Senter and Smith, 1967) and SMOG (McLaughlin, 1969), to automatic machine learning approaches (e.g., Pe-

tersen and Ostendorf, 2009; Crossley *et al.*, 2011). These methods measure general, objective difficulty and cannot adapt to each user's subjective perception of text difficulty. The commonly used classifiers mostly do binary difficulty assessment: either the document is difficult (intended for professionals) or not. Those automatic versions of the readability formulas that use language-specific preprocessing or specific word lists are difficult to extend to new languages and domains. Moreover, automatic readability formulas face the same restriction with the traditional ones: they are usually intended for pupils and thus cannot be directly applied to assessing expertise domain texts that are intended for adults, because the scale does not reach the expert adult level. The readability and text difficulty measures in the literature aim to analyse the difficulty of a text to the reader, but very rarely specific user modelling approaches have been proposed. An exception is Liu *et al.* (2004) who analysed short search engine queries to recognise the reading level of the user.

METHOD A novel user-specific text difficulty measure was proposed in Publication VII. The method enables, for instance, offering information in a personalised manner based on the user's knowledge of different domains. The method compares terms appearing in a document and terms known by the user. The terms are extracted automatically from text using the *Likey* method (Publication I). Two ways to collect information about what terms the user knows are presented in Publication VII: by directly asking the users the difficulty of terms or, as a novel automatic approach, indirectly by analysing texts written by the users.

The information about the text to be analysed is collected to a document vector and the knowledge of a user to a user vector. A *document vector* $\mathbf{d}_j$ of document j, $j = \{1,\ldots,D\}$ has counts of unigram terms $c_j(t_i)$, $t_i \in \mathcal{T}$, $i \in \{1,\ldots,F\}$, as its elements

$$\mathbf{d}_j = [c_j(t_1), c_j(t_2), \ldots, c_j(t_F)], \tag{16}$$

where F is the number of features in the document set feature space $\mathcal{T}$. Each individual user $\mathbf{u}_k$, $k = \{1,\ldots,U\}$, has a *user profile vector* in the same feature space $\mathcal{T}$ with the document vectors. The values for each term t_i in the user profile vector correspond to *term difficulty* $\theta(t_i) \in [0,1]$, the perceived difficulty of the term by the user

$$\mathbf{u}_k = [\theta_k(t_1), \theta_k(t_2), \ldots, \theta_k(t_F)]. \tag{17}$$

Given document vector $\mathbf{d}_j$ and user vector $\mathbf{u}_k$, the *user-specific difficulty* of a document δ_{du} is defined in the following way:

$$\delta_{du}(\mathbf{d}_j, \mathbf{u}_k) = \frac{1}{M_j} \cdot \mathbf{u}_k \mathbf{d}_j^{\mathrm{T}}, \tag{18}$$

where the document vector is normalised with M_j, the length of document j, i.e., the total number of words in document j. The *difficulty of a set of documents* $\mathbf{d}$ for a certain user $\mathbf{u}_k$ can be defined as

$$\delta_u(\mathbf{d}, \mathbf{u}_k) = \frac{1}{N} \sum_{j=1}^{N} \frac{1}{M_j} \mathbf{u}_k \mathbf{d}_j^{\mathrm{T}}, \tag{19}$$

where N is the total number of documents in the document set. The *general difficulty of document* $\mathbf{d}_j$ for a set of users $\mathbf{u}$ is defined as

$$\delta_d(\mathbf{d}_j, \mathbf{u}) = \frac{1}{U} \sum_{k=1}^{U} \frac{1}{M_j} \mathbf{u}_k \mathbf{d}_j^{\mathrm{T}}, \tag{20}$$

where U is the number of users who participate in the term difficulty rating. The *general difficulty of document set* $\mathbf{d}$ for a set of users $\mathbf{u}$ is defined as

$$\delta(\mathbf{d}, \mathbf{u}) = \frac{1}{UN} \sum_{k=1}^{U} \sum_{j=1}^{N} \frac{1}{M_j} \mathbf{u}_k \mathbf{d}_j^{\mathrm{T}}. \tag{21}$$

The direct and indirect term difficulty rating approaches have different ways of collecting the user vectors. In the *direct approach*, users were asked the difficulty of a list of keyphrases, ranging from 0 (very easy) to 3 (very difficult). The human-rated difficulties were scaled between 0 and 1 by dividing by 3. In the *indirect approach*, the information about the familiarity of terminology was collected from texts the users had written: dissertations and blogs. The frequencies of the document keyphrases $c(t)$ were calculated in these user texts and transformed with a simple continuous squashing function between 0 and 1 to obtain term difficulty

$$\theta(t) = 1 - \tanh(c(t) + 1/3). \tag{22}$$

RESULTS In the evaluation of the proposed method, a medical document collection in Finnish was used. The document collection consists of four document sets about diseases and their treatment: one (called *ProPro*) is intended for professionals and three other sets (*ProLay*, *Lay1* and *Lay2*) for lay people. More details about the data can be found in Publication VII.

The results of the direct approach for user modelling with human-rated terms were very good, see Table 21. When further comparing the results to six traditional readability measure baselines, five out of six baselines agreed with the proposed difficulty measure, namely, that the *ProPro* set indeed is the most difficult set out of the four document sets.

Table 21: Direct approach: General difficulty δ of a document set intended for professionals (*ProPro*) and three document sets for lay people (*ProLay*, *Lay1* and *Lay2*).

Document set	δ
ProPro	**0.00302**
ProLay	0.00198
Lay1	0.00152
Lay2	0.00153

Similarly, the user-specific difficulties δ_u of the document sets by an individual lay user, calculated with Equation 19, are displayed in Table 22. The results indicate that the difficulty of the *ProPro* set for this non-medical expert user is noticeable compared to the lay documents.

Table 22: Direct approach: User-specific difficulty δ_u of the document sets, from the viewpoint of an individual lay user. The *ProPro* set is intended for professionals and the other sets for lay people.

Document set	δ_u
ProPro	**0.00575**
ProLay	0.00340
Lay1	0.00285
Lay2	0.00309

The user-specific difficulties of individual documents δ_{du} in the four data sets were calculated and the documents were ordered according to the descending difficulty. Histograms of the documents are shown in Figure 13. As a result, the professional *ProPro* documents are concentrated at the beginning (the most difficult part) of the ordered document list.

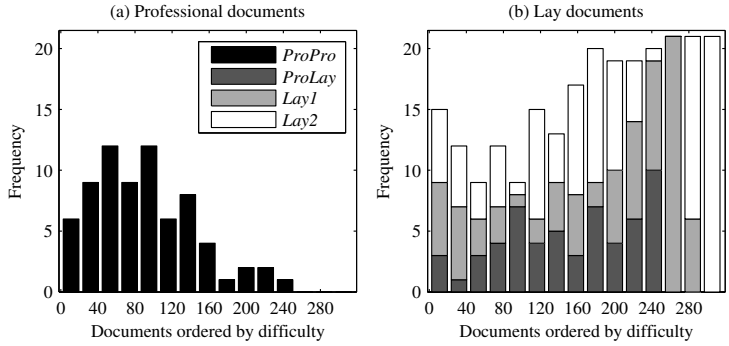


Figure 13: Direct approach: Histograms of (a) professional and (b) lay documents in the ordered list according to the difficulty δ_{du} of each document, perceived by an individual lay user. The most difficult documents are on the left side of the diagrams. From Publication VII.

In the indirect approach, texts written by users are analysed automatically to obtain the user vectors. A collection of dissertations on the medical domain serve as texts from professional users and blog texts as texts from lay users. The results for extracting $\varphi = 3$ keyphrases per 100 words in a document and measuring general difficulty (Equation 21) are given in Table 23. The document set *ProPro* intended for professional users was more difficult than the other sets for both professional (Dissertations) and lay users (Blogs). Furthermore, lay users considered all the texts more difficult than did professional users.

The results show that the method is able to distinguish between documents written for lay people and documents written for experts. The experiments were conducted for the Finnish language but the method is also applicable to other languages: the difficulty measure is based on *Likey* which has been successfully applied to several other languages (Publication I). Especially the indirect approach is easily applicable to new languages and domains, because the

Table 23: Indirect approach: General difficulty δ of a document set intended for professionals (*ProPro*) and three document sets for lay people (*ProLay*, *Lay1* and *Lay2*), with $\varphi = 3$. The user groups are professional (Dissertations), lay users (Blogs) and all users. The most difficult set is shown in bold face for each user group, and significant differences between the professional and the lay sets are underlined.

Document set	δ**(Dissertations)**	δ**(Blogs)**	δ**(all)**
ProPro	**0.093**	**0.095**	**0.094**
ProLay	0.086	0.090	0.088
Lay1	0.089	0.092	0.091
Lay2	0.086	0.089	0.088

only language-specific components were a reference corpus and a rule-based stemmer which could be replaced with a statistical stemmer, for example *Morfessor* (Creutz and Lagus, 2007). More details of the method and further results can be found in Publication VII.

4.6 OTHER NLP TASKS WITH LANGUAGE-INDEPENDENT APPLICATIONS

In this dissertation, two major NLP tasks, information retrieval (IR) and statistical machine translation (SMT), which can be addressed with language-independent methods, have been discussed just on the level of a mention. These topics are briefly discussed in this section. Multilinguality has a significant role in both IR and SMT. Many IR systems index documents in multiple languages and also perform retrieval across languages. A statistical machine translation system plays with at least two languages – usually many more. Multilingual corpora which can be used for both training and evaluating these systems were discussed earlier in Section 2.3.

4.6.1 *Information retrieval*

Information retrieval (IR) is an NLP task of retrieving a set of documents from a large document base as a response to a query given by a user. Many IR approaches are based on the vector space model (Salton *et al.*, 1975), discussed earlier in Section 2.2.4, which is a highly language-independent approach. However, here as well as in many other NLP approaches, language-specific information, such as stop word lists, stemming or lemmatizing, and part-of-speech tags may be, and are, used for improving the accuracy. For a textbook description, see Manning *et al.* (2008).

PageRank is the Web page ranking method used by the Google Web page search engine. It counts the number of inlinks and outlinks to and from each page (Brin and Page, 1998). *PageRank* is used together with the page text contents and anchors (link texts on other pages) to select the best matching pages for a query. This method has has become an extremely popular search engine and has shown its applicability across domains, genres and languages.

Cross-lingual IR

There is a substantial amount of research in the *cross-lingual* setting of IR. Multilingual parallel corpora make a variety of approaches possible. For example, an approach by Dumais *et al.* (1997) used a set of aligned bilingual (translated) documents and created a vector space with *latent semantic indexing*. From the vector space it is possible to retrieve documents in any of the two languages without the need to translate the queries. In a related task, *mate retrieval*, a document in a source language is used as the query and the corresponding document (mate) in a target language is considered to be the only relevant document to the query. Studies have been conducted for example with the English–French (Vinokourov *et al.*, 2003), Japanese–English (Li and Shawe-Taylor, 2007), and English–Spanish language pairs (Hardoon and Shawe-Taylor, 2007). Chew and Abdelali (2007) showed that cross-lingual retrieval precision increases when more parallel languages are added to the vector space. In their experiments, 31 distinct languages were used, consisting of 47 parallel translations.

4.6.2 *Statistical machine translation*

Statistical machine translation (SMT) is an automatic translation task, in which, as opposed to rule-based machine translation, the translation pairs are learned automatically from a text corpus. Bilingual or multilingual parallel corpora are required between the source and target languages to learn the translations. One of the main advantages of statistical machine translation is the possibility of a very quick extension to new languages: the change simply needs a new bilingual corpus and automatic training of the unsupervised system. SMT has a relatively low out-of-vocabulary (OOV) rate compared to rule-based translation, thanks to the large corpora used. Drawbacks of SMT include semantically incorrect translations which are rather common. A textbook description of statistical machine translation can be found in Koehn (2010).

Statistical machine translation is one of the NLP tasks that can be approached with methods of high independence of the used language. The translation procedure starts with aligning the bilingual corpus on a sentence level and further aligning words and phrases, using, e.g., *GIZA++* (Och and Ney, 2000). A full-range phrase-based statistical machine translation system *Moses* (Koehn *et al.*, 2007) was built in 2006. It has been further developed and it is also the basis for many other machine translation systems. Virpioja *et al.* (2007) experimented with unsupervised morpheme segmentation and *Moses*, and found that morpheme segmentation decreased the number of OOV words but did not help to increase the *BLEU* evaluation score. Besides morpheme segmentation, also other NLP approaches have been attached to the translation system, for example lexical substitution (Bangalore *et al.*, 2007) and word sense disambiguation (Carpuat and Wu, 2007).

5

SUMMARY AND CONCLUSIONS

This thesis contributed to three themes within the field of natural language processing (NLP): *language independence* which usually also involves the independence of the domain, *subjectivity in language use* that takes the variations between human beings into account, and *fully automatic methods* that enable quick adoption of a method to a new environment. About one half of the research presented in this thesis is related in one way or another to a language-independent keyphrase extraction method *Likey* that can be seen, depending on the viewpoint and application, as a method for keyphrase extraction, feature selection or dimensionality reduction. In addition to the language independence, the method is also fully automatic by not requiring any manual intervention. *Likey* was tested on documents in 11 European languages, using different evaluation methods. It was further applied to feature selection for taxonomy learning (Finnish, English, Spanish) and text difficulty assessment. The latter contributed also to the theme of subjective language use. This dissertation further contributed to a common NLP task, document clustering, for which differences between dimensionality reduction and distance measures were tested (for English and Hindi; with the state-of-the-art preprocessing) and an evaluation method for document representation was presented (using Danish, English, Finnish, German and Swedish). In contrast to the approaches of very high level of language independence above, the last contribution to a task of near-synonym lexical choice is with a take-whatever-linguistic-features-available approach, ending up with an extensive set of the linguistic features and using them with various machine learning methods.

The language- and domain-independent methods discussed in this thesis would give a considerable benefit in many research areas which nowadays collect data manually from textual resources, possibly in multiple languages: clustering and other unsupervised methods can be applied to explore topics, relations and structures in large document collections. This can be done for example in comparing domains and languages from the viewpoint of actual language usage (in the research field of linguistics), the semantic associations (psychology), or changes over decades and centuries (history). Language-independent methods are flexible and applicable to many domains and are thus an attractive alternative to labour-intensive manual encoding of linguistic knowledge. Besides research, also many business areas dealing with unstructured textual data would find the methods in this thesis valuable: some examples would be customer relationship management, content management and master data management. In an international company having documents in multiple languages, the methods presented in this thesis would help avoid stress and inconvenience: For example, marketing departments having large amounts of data about their customers and even more data available online, could organize, an-

alyze and visualize their unstructured multilingual data, applying the same text mining methods world-wide.

Many language-independent methods perform rather well on their own, but in order to increase the performance, some language-specific components are needed. This may be one of the reasons why the number of published fully language-independent studies is still so small. Machine learning methods can be compared to an extremely good language learner, a child, who does not need a grammar (a language-specific component) to learn a language, but needs language data in its actual use (unsupervised learning), often has also a teacher (supervised learning) and, in addition, gets a lot of grounded information in the world. From this viewpoint of human learning, it has to be said that with unsupervised learning approaches, there is perhaps not enough information to learn the *meaning* of a word or a sentence from a collection of texts only. Thus, some additional language-specific components would be needed until the meanings can be automatically extracted from grounded multimodal data, such as pictures, videos or data obtained through other sensors.

The research in this dissertation has raised interest in several related topics and further research directions could be within the following: (1) the use of language universals in language-independent methods, (2) automatic discovery of language universals from large multilingual corpora of hundreds or thousands of languages, and (3) machine translation with several source languages, following the idea by Chew and Abdelali (2007) who found out that adding few languages decreases the performance of cross-lingual information retrieval, but adding even more languages yields significantly better results.

Today, a system that is able to analyze texts in all the about 7000 languages in the world is only a dream. First, collecting such amounts of clean data would require a remarkable effort. The Universal Declaration of Human Rights and the Bible are good starting points with their 300–400 languages. In addition, some evaluation data would be needed in order to measure the system performance automatically. Apparently, such a huge-scale system would require a language-independent pre-analysis process to group languages according to their characteristics and to transform each language to a uniform easily-processed format, involving, for example, word boundary detection in languages like Chinese and Japanese, and morphology segmentation in the group of languages including Finnish and Turkish. Having this view in mind, it is clear that there are still many things to do within the area of language independence and the entire field of natural language processing: The computers and robots are not yet ready to talk like us, and they are not likely to replace our communication and language skills in the near future.

Gediminas Adomavicius and Alexander Tuzhilin, 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, 17(6):734–749.

Eneko Agirre, Lluís Màrquez, and Richard Wicentowski, editors, 2007. *Proceedings of the Fourth International Workshop on Semantic Evaluation (SemEval)*. Association for Computational Linguistics, Prague, Czech Republic.

Eneko Agirre, Oier Lopez de Lacalle, Christiane Fellbaum, Andrea Marchetti, Antonio Toral, and Piek Vossen, 2009. SemEval-2010 Task 17: All-words word sense disambiguation on a specific domain. In *Proceedings of the NAACL HLT Workshop on Semantic Evaluations: Recent Achievements and Future Directions (SEW-2009)*, pages 123–128, Boulder, CO, USA. Association for Computational Linguistics.

Sameh Alansary, Magdy Nagi, and Noha Adly, 2006. Processing Arabic text content: The encoding component in an interlingual system for man-machine communication in natural language. In *Proceedings of the 6th International Conference on Language Engineering*, Cairo, Egypt.

Ethem Alpaydin, 2010. *Introduction to Machine Learning*. The MIT Press, Cambridge, MA, USA, 2nd edition.

Robert A. Amsler, 1981. A taxonomy for English nouns and verbs. In *Proceedings of the 19th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 133–138, Standford, CA, USA. Association for Computational Linguistics.

Elaine S. Andersen, 1978. Lexical universals of body-part terminology. In J. H. Greenberg, C. A. Ferguson, and E. A. Moravcsik, editors, *Universals of Human Language*, volume 3, pages 335–368. Stanford University Press, Stanford, CA, USA.

Marianna Apidianaki, 2009. Data-driven semantic analysis for multilingual WSD and lexical selection in translation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 77–85, Athens, Greece. Association for Computational Linguistics.

Liliana Ardissono, Luca Console, and Ilaria Torre, 2001. An adaptive system for the personalized access to news. *Artificial Intelligence Communications*, 14(3):129–148.

Shlomo Argamon, Marin Šarić, and Sterling S. Stein, 2003. Style mining of electronic messages for multiple authorship discrimination: first results. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge*

Discovery and Data Mining (KDD'03), pages 475–480, Washington, D.C., USA. ACM.

Antti Arppe. 2008. *Univariate, bivariate, and multivariate methods in corpus-based lexicography – a study of synonymy*. PhD thesis, University of Helsinki, Finland. http://urn.fi/URN:ISBN:978-952-10-5175-3.

Francis R. Bach and Michael I. Jordan, 2003. Kernel independent component analysis. *Journal of Machine Learning Research*, 3:1–48.

Amit Bagga and Breck Baldwin, 1998. Entity-based cross-document coreferencing using the vector space model. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (Coling-ACL '98)*, volume 1, pages 79–85, Montreal, Canada. Association for Computational Linguistics.

Srinivas Bangalore, Patrick Haffner, and Stephan Kanthak, 2007. Statistical machine translation through global lexical selection and sentence reconstruction. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 152–159, Prague, Czech Republic. Association for Computational Linguistics.

Leonard E. Baum, Ted Petrie, George Soules, and Norman Weiss, 1970. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, 41(1):164–171.

Emily M. Bender, 2009. Linguistically naïve!= language independent: Why NLP needs linguistic typology. In *Proceedings of the EACL 2009 Workshop on the Interaction between Linguistics and Computational Linguistics: Virtuous, Vicious or Vacuous?*, pages 26–32, Athens, Greece. Association for Computational Linguistics.

Emily M. Bender, 2011. On achieving and evaluating language independence in NLP. *Linguistic Issues in Language Technology*, 6(3).

Dominik Benz and Andreas Hotho, 2007. Position paper: Ontology learning from folksonomies. In A. Hinneburg, editor, *LWA 2007: Lernen - Wissen - Adaption, Workshop Proceedings*, pages 109–112, Halle–Wittenberg, Germany. Martin-Luther-University.

Jean-Paul Benzécri, 1973. *L'Analyse des Données. Volume II. L'Analyse des Correspondances (in French)*. Dunod, Paris, France.

Brent Berlin and Paul Kay, 1969. *Basic Color Terms: Their Universality and Evolution*. University of California Press, Berkeley, CA, USA.

John R. L. Bernard, editor, 1990. *The Macquarie Encyclopedic Thesaurus*. The Macquarie Library, Sydney, Australia.

Delphine Bernhard, 2008. Simple morpheme labelling in unsupervised morpheme analysis. In C. Peters, V. Jijkoun, T. Mandl, H. Müller, D. W. Oard,

A. Peñas, V. Petras, and D. Santos, editors, *Advances in Multilingual and Multimodal Information Retrieval*, volume 5152 of *Lecture Notes in Computer Science*, pages 873–880. Springer-Verlag, Berlin/Heidelberg, Germany.

Romaric Besançon and Martin Rajman, 2002. Evaluation of a vector space similarity measure in a multilingual framework. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC 2002)*, pages 1537–1542, Las Palmas, Spain. European Language Resources Association.

Chris Biemann, 2005. Ontology learning from text: A survey of methods. *GLDV – Journal for Computational Linguistics and Language Technology*, 20(2):75–93.

Daniel Billsus and Michael J. Pazzani, 2000. User modeling for adaptive news access. *User Modeling and User-Adapted Interaction*, 10(2-3):147–180.

Ella Bingham and Heikki Mannila, 2001. Random projection in dimensionality reduction: applications to image and text data. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2001)*, pages 245–250, San Francisco, CA, USA. ACM.

Christopher M. Bishop, 2006. *Pattern recognition and machine learning*. Springer, New York, NY, USA.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan, 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

John Blitzer, Ryan McDonald, and Fernando Pereira, 2006. Domain adaptation with structural correspondence learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 120–128, Sydney, Australia. Association for Computational Linguistics.

Avrim Blum and Tom Mitchell, 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the Eleventh Annual Conference on Computational learning theory (COLT)*, pages 92–100, Madison, WI, USA. ACM.

Frank Boas, 1966. *Introduction to the Handbook of American Indian Languages*, volume 1. Smithsonian Institution, Washington, D.C., USA.

Georgeta Bordea and Paul Buitelaar, 2010. DERIUNLP: A context based approach to automatic keyphrase extraction. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*, pages 146–149, Uppsala, Sweden. Association for Computational Linguistics.

Magnus Borga. 1998. *Learning Multidimensional Signal Processing*. PhD thesis, Linköping University, Sweden.

David B. Bracewell, Fuji Ren, and Shingo Kuriowa, 2005. Multilingual single document keyword extraction for information retrieval. In *Proceedings of the 2005 IEEE International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE'05)*, pages 517–522, Wuhan, China. IEEE.

Sergey Brin and Lawrence Page, 1998. The anatomy of a large-scale hypertextual Web search engine. *Computer networks and ISDN systems*, 30(1–7):107–117.

Peter F. Brown, Peter V. deSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai, 1992. Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479.

Cecil H. Brown, 1976. General principles of human anatomical partonomy and speculations on the growth of partonomic nomenclature. *American Ethnologist*, 3(3):400–424.

Alexander Budanitsky and Graeme Hirst, 2006. Evaluating WordNet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47.

John A. Bullinaria and Joseph P. Levy, 2007. Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, 39:510–526.

John A. Bullinaria and Joseph P. Levy, 2012. Extracting semantic representations from word co-occurrence statistics: stop-lists, stemming, and SVD. *Behavior Research Methods*, 44.

Jean-François Cardoso, 1998. Blind signal separation: statistical principles. *Proceedings of the IEEE*, 9(10):2009–2025.

Francesca Carmagnola, Federica Cena, Luca Console, Omar Cortassa, Cristina Gena, Anna Goy, Ilaria Torre, Andrea Toso, and Fabiana Vernero, 2008. Tag-based user modeling for social multi-device adaptive guides. *User Modeling and User-Adapted Interaction*, 18(5):497–538.

Marine Carpuat and Dekai Wu, 2007. Improving statistical machine translation using word sense disambiguation. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 61–72, Prague, Czech Republic.

Zheng Chen and Heng Ji, 2010. Graph-based clustering for computational linguistics: A survey. In *Proceedings of the 2010 Workshop on Graph-based Methods for Natural Language Processing (TextGraphs-5)*, pages 1–9, Uppsala, Sweden. Association for Computational Linguistics.

Peter A. Chew and Ahmed Abdelali, 2007. Benefits of the 'massively parallel Rosetta stone': Cross-language information retrieval with over 30 languages. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 872–879, Prague, Czech Republic. Association for Computational Linguistics.

Peter A. Chew, Steve J. Verzi, Travis L. Bauer, and Jonathan T. McClain, 2006. Evaluation of the Bible as a resource for cross-language information retrieval.

In *Proceedings of the Workshop on Multilingual Language Resources and Interoperability*, pages 68–74, Sydney, Australia. Association for Computational Linguistics.

Kenneth W. Church and William A. Gale, 1991. Concordances for parallel text. In *Proceedings of the Seventh Annual Conference of the UW Centre for the New OED and Text Research*, pages 40–62, Oxford, UK.

Philipp Cimiano, Andreas Hotho, and Steffen Staab, 2005. Learning concept hierarchies from text corpora using formal concept analysis. *Journal of Artificial Intelligence Research*, 24(1):305–339.

Alexander Clark, 2000. Inducing syntactic categories by context distribution clustering. In C. Cardie, W. Daelemans, C. Nedellec, and E. Tjong Kim Sang, editors, *Proceedings of the Fourth Conference on Computational Language Learning (CoNLL-2000) and the Second Learning Language in Logic Workshop (LLL)*, pages 91–94, Lisbon, Portugal. Association for Computational Linguistics.

K. Robert Clarke, Paul J. Somerfield, and M. Gee Chapman, 2006. On resemblance measures for ecological studies, including taxonomic dissimilarities and a zero-adjusted Bray-Curtis coefficient for denuded assemblages. *Experimental Marine Biology and Ecology*, 330(1):55–80.

Pierre Comon, 1994. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314.

Silvia Coradeschi and Alessandro Saffiotti, 2000. Anchoring symbols to sensor data: preliminary report. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-00)*, pages 129–135, Austin, TX, USA. AAAI/The MIT Press.

Malcolm Coulthard, 2004. Author identification, idiolect, and linguistic uniqueness. *Applied linguistics*, 25(4):431–447.

Thomas M. Cover and Peter E. Hart, 1967. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27.

Mathias Creutz and Krista Lagus, 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Transactions on Speech and Language Processing*, 4(1):Article 3.

Mathias Creutz. 2006. *Induction of the morphology of natural language*. PhD thesis, Helsinki University of Technology, Espoo, Finland.

William Croft and D. Alan Cruse, 2004. *Cognitive linguistics*. Cambridge University Press, Cambridge, UK.

Scott A. Crossley, Jerry Greenfield, and Danielle S. McNamara, 2008. Assessing text readability using cognitively based indices. *Tesol Quarterly*, 42(3):475–493.

Scott A. Crossley, David B. Allen, and Danielle S. McNamara, 2011. Text readability and intuitive simplification: A comparison of readability formulas. *Reading in a Foreign Language*, 23(1):84–101.

Silviu Cucerzan and David Yarowsky, 1999. Language independent named entity recognition combining morphological and contextual evidence. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing (EMNLP) and Very Large Corpora (VLC)*, pages 90–99, College Park, MD, USA.

Marc Damashek, 1995. Gauging similarity with n-grams: Language-independent categorization of text. *Science*, 267:843–848.

Fred Damerau, 1993. Generating and evaluating domain-oriented multi-word terms from text. *Information Processing and Management*, 29(4):433–447.

Hal Daumé III, Abhishek Kumar, and Avishek Saha, 2010. Frustratingly easy semi-supervised domain adaptation. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, pages 53–59, Uppsala, Sweden. Association for Computational Linguistics.

Hal Daumé III, 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 256–263, Prague, Czech Republic. Association for Computational Linguistics.

Ernesto D'Avanzo. 2005. *Using Keyphrases for Text Mining: Applications and Evaluation*. PhD thesis, Department of Information and Communication Sciences, University of Trento, Trento, Italy.

Olivier Y. de Vel, Alison Anderson, Malcolm Corney, and George M. Mohay, 2001. Mining e-mail content for author identification forensics. *ACM Sigmod Record*, 30(4):55–64.

Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman, 1990. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407.

Klaas Dellschaft and Steffen Staab, 2006. On how to perform a gold standard based evaluation of ontology learning. In I. F. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, and L. Aroyo, editors, *The Semantic Web - ISWC 2006*, volume 4273 of *Lecture Notes in Computer Science*, pages 228–241. Springer-Verlag, Berlin/Heidelberg, Germany.

Lars-Olof Delsing and Katarina Lundin Åkesson, 2005. *Håller språket ihop Norden? En forskningsrapport om ungdomars förståelse av dansa, svensa och norska (in Swedish)*. Number 573 in TemaNord. Nordiska ministerrådet, Köpenhamn, Denmark.

Michel M. Deza and Elena Deza, 2009. *Encyclopedia of distances*. Springer-Verlag, Berlin/Heidelberg, Germany.

Michael Dittenbach, Dieter Merkl, and Andreas Rauber, 2000. The growing hierarchical self-organizing map. In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN 2000)*, volume 6, pages 15–19, Como, Italy. IEEE.

Marcus Dobrinkat and Jaakko J. Väyrynen, 2010. Experiments with domain adaptation methods for statistical MT: From European Parliament proceedings to Finnish newspaper text. In *Proceedings of the 14th Finnish Artificial Intelligence Conference (STeP 2010)*, pages 31–38, Espoo, Finland. Finnish Artificial Intelligence Society.

Susan T. Dumais, Todd A. Letsche, Michael L. Littman, and Thomas K. Landauer, 1997. Automatic cross-language retrieval using latent semantic indexing. In *Proceedings of the AAAI Spring Symposium on Cross-Language Text and Speech Retrieval*, pages 15–21, Standford, CA, USA. The AAAI Press.

Susan T. Dumais, 1991. Improving the retrieval of information from external sources. *Behavior Research Methods, Instruments, & Computers*, 23(2):229–236.

Ted Dunning, 1993. Accurate methods for the statistics of surprise and coincidence. *Computational linguistics*, 19(1):61–74.

Nicholas D. Duran, Cedrick Bellissens, Roger S. Taylor, and Danielle S. McNamara, 2007. Quantifying text difficulty with automated indices of cohesion and semantics. In *Proceedings of the 29th Annual Meeting of the Cognitive Science Society (CogSci)*, pages 233–238, Nashville, TN, USA. Cognitive Science Society.

Jennifer G. Dy and Carla E. Brodley, 2004. Feature selection for unsupervised learning. *Journal of Machine Learning Research*, 5:845–889.

Philip Edmonds, 1997. Choosing the word most typical in context using a lexical co-occurrence network. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 507–509, Madrid, Spain. Association for Computational Linguistics.

Samhaa R. El-Beltagy and Ahmed Rafea, 2009. KP-Miner: A keyphrase extraction system for English and Arabic documents. *Information Systems*, 34(1):132–144.

Samhaa R. El-Beltagy and Ahmed Rafea, 2010. KP-Miner: Participation in SemEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*, pages 190–193, Uppsala, Sweden. Association for Computational Linguistics.

Katrin Erk and Carlo Strapparava, editors, 2010. *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*. Association for Computational Linguistics, Uppsala, Sweden.

Güneş Erkan and Dragomir R. Radev, 2004. LexPageRank: Prestige in multi-document text summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 365–371, Barcelona, Spain. Association for Computational Linguistics.

Gerard Escudero, Lluís Màrquez, and German Rigau, 2000. An empirical study of the domain dependence of supervised word sense disambiguation systems. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000)*, pages 172–180, Hong Kong, China. Association for Computational Linguistics.

Francesca Fallucchi and Fabio Massimo Zanzotto, 2011. Inductive probabilistic taxonomy learning using singular value decomposition. *Natural Language Engineering*, 17(1):71–94.

Meagan T. Farrell, Lise Abrams, and Katherine K. White, 2012. The role of priming in lexical access and speech production. In N. Hsu and Z. Schütt, editors, *Psychology of Priming*, chapter 10, pages 205–244. Nova Science Publishers, New York, NY, USA.

David Faure and Claire Nédellec, 1998. A corpus-based conceptual clustering method for verb frames and ontology acquisition. In *Proceedings of the LREC workshop on Adapting Lexical and Corpus Resources to Sublanguages and Applications*, pages 5–12, Granada, Spain.

Christiane Fellbaum, editor, 1998. *WordNet: An Electronic Lexical Database*. The MIT Press, Cambridge, MA, USA.

Yansong Feng and Mirella Lapata, 2008. Automatic image annotation using auxiliary text information. *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT)*, pages 272–280.

Geoffrey Finch, 2003. *How to study linguistics: A guide to understanding language*. Palgrave Study Guides: Literature. Palgrave Macmillan, 2nd edition.

Aidan Finn and Nicholas Kushmerick, 2006. Learning to classify documents according to genre. *Journal of the American Society for Information Science and Technology*, 57(11):1506–1518.

Eibe Frank, Gordon W. Paynter, Ian H. Witten, Carl Gutwin, and Craig G. Nevill-Manning, 1999. Domain-specific keyphrase extraction. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI 99)*, pages 668–673, Stockholm, Sweden.

George W. Furnas, Thomas K. Landauer, Louis M. Gomez, and Susan T. Dumais, 1987. The vocabulary problem in human-system communication. *Communications of the ACM*, 30(11):964–971.

Evgeniy Gabrilovich and Shaul Markovitch, 2007. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07)*, pages 1606–1611, Hyderabad, India.

William A. Gale, Kenneth W. Church, and David Yarowsky, 1992. One sense per discourse. In *Proceedings of the 4th DARPA Speech and Natural Language Workshop*, pages 233–237. Association for Computational Linguistics.

Éric Gaussier, Jean-Michel Renders, Irina Matveeva, Cyril Goutte, and Hervé Déjean, 2004. A geometric view on bilingual lexicon extraction from comparable corpora. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 526–533, Barcelona, Spain. Association for Computational Linguistics.

Sharon Goldwater and Thomas L. Griffiths, 2007. A fully Bayesian approach to unsupervised part-of-speech tagging. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 744–751, Prague, Czech Republic. Association for Computational Linguistics.

José Manuel Gómez-Soriano, Manuel Montes-y-Gómez, Emilio Sanchis-Arnal, Luis Villaseñor-Pineda, and Paolo Rosso, 2005. Language independent passage retrieval for question answering. In A. Gelbukh, A. de Albornoz, and H. Terashima, editors, *MICAI 2005: Advances in Artificial Intelligence*, volume 3789 of *Lecture Notes in Computer Science*, pages 816–823. Springer-Verlag, Berlin/Heidelberg, Germany.

Raymond G. Gordon and Barbara F. Grimes, editors, 2005. *Ethnologue: Languages of the world*. SIL International, Dallas, TX, USA, 15th edition.

Olof Görnerup and Jussi Karlgren, 2010. Cross-lingual comparison between distributionally determined word similarity networks. In *Proceedings of the 2010 Workshop on Graph-based Methods for Natural Language Processing (TextGraphs-5)*, pages 48–54, Uppsala, Sweden. Association for Computational Linguistics.

Natalia Grabar and Sonia Krivine, 2007. Application of cross-language criteria for the automatic distinction of expert and non expert online health documents. In R. Bellazzi, A. Abu-Hanna, and J. Hunter, editors, *Artificial Intelligence in Medicine*, volume 4594 of *Lecture Notes in Computer Science*, pages 252–256. Springer-Verlag, Berlin/Heidelberg, Germany.

Arthur C. Graesser, Danielle S. McNamara, Max M. Louwerse, and Zhiqiang Cai, 2004. Coh-metrix: Analysis of text on cohesion and language. *Behavioral Research Methods, Instruments, and Computers*, 36(2):193–202.

Joseph H. Greenberg, 1966. *Language Universals: With Special Reference to Feature Hierarchies*. Mouton de Gruyter, Berlin.

George Grefenstette, 1994. *Explorations in Authomatic Thesaurus Construction*. Kluwer Academic Publishers, Boston.

Carl Gutwin, Gordon Paynter, Ian Witten, Craig Nevill-Manning, and Eibe Frank, 1999. Improving browsing in digital libraries with keyphrase indexes. *Decision Support Systems*, 27(1–2):81–104.

Aria Haghighi and Dan Klein, 2006. Prototype-driven learning for sequence models. In *Proceedings of the Human Language Technology Conference of the NAACL (HLT/NAACL-04)*, pages 320–327, New York, NY, USA. Association for Computational Linguistics.

Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein, 2008. Learning bilingual lexicons from monolingual corpora. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT)*, pages 771–779, Columbus, OH, USA. Association for Computational Linguistics.

Khaled M. Hammouda, Diego N. Matute, and Mohamed S. Kamel, 2005. Corephrase: Keyphrase extraction for document clustering. In P. Perner and A. Imiya, editors, *Machine Learning and Data Mining in Pattern Recognition*, volume 3587 of *Lecture Notes in Computer Science*, pages 265–274. Springer-Verlag, Berlin/Heidelberg, Germany.

David R. Hardoon and John Shawe-Taylor. 2007. Sparse canonical correlation analysis. Technical report, University College London, London, UK.

David R. Hardoon, Sandor Szedmak, and John Shawe-Taylor, 2004. Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 16(12):2639–2664.

Trevor A. Harley, 1995. *The Psychology of Language: From Data to Theory*. Psychology Press, East Sussex, UK.

Harry Horace. Harman, 1960. *Modern Factor Analysis*. University of Chicago Press, Chicago, IL, USA.

Stevan Harnad, 1990. The symbol grounding problem. *Physica D*, 42(1–3):335–346.

Simon Haykin, 1994. *Neural networks: a comprehensive foundation*. Prentice Hall, PTR Upper Saddle River, NJ, USA.

Riikka Henriksson, Tomi Kauppinen, and Eero Hyvönen, 2008. Core geographical concepts: Case Finnish geo-ontology. In S. Boll, C. Jones, E. Kansa, P. Kishor, M. Naaman, R. Purves, A. Scharl, and E. Wilde, editors, *Proceedings of the First International Workshop on Location and the Web (LocWeb 2008)*, volume 300 of *ACM International Conference Proceedings Series*, pages 57–60, Beijing, China. ACM.

Charles F. Hockett, 1963. The problem of universals in language. In J. H. Greenberg, editor, *Universals of Language*, pages 1–29. The MIT Press, Cambridge, MA, USA.

Thomas Hofmann, 1999. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence (UAI'99)*, pages 289–296, Stockholm, Sweden. Morgan Kaufmann Publishers.

Timo Honkela, Ville Pulkki, and Teuvo Kohonen, 1995. Contextual relations of words in Grimm tales, analyzed by self-organizing map. In F. Fogelman-Soulié and P. Gallinari, editors, *Proceedings of the International Conference on Artificial Neural Networks (ICANN'95)*, volume II, pages 3–7, Paris, France.

Timo Honkela, Ville Könönen, Tiina Lindh-Knuutila, and Mari-Sanna Paukkeri, 2008. Simulating processes of concept formation and communication. *Journal of Economic Methodology*, 15(3):245–259.

Timo Honkela, Aapo Hyvärinen, and Jaakko J. Väyrynen, 2010. WordICA—emergence of linguistic representations for words by independent component analysis. *Natural Language Engineering*, 16:277–308.

Harold Hotelling, 1933. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441.

Harold Hotelling, 1936. Relations between two sets of variates. *Biometrika*, 28(3):321–377.

Anna Huang, 2008. Similarity measures for text document clustering. In *Proceedings of the New Zealand Computer Science Research Student Conference (NZCSRSC 2008)*, pages 49–56, Christchurch, New Zealand.

Aapo Hyvärinen and Erkki Oja, 1997. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–1492.

Aapo Hyvärinen, Juha Karhunen, and Erkki Oja, 2001. *Independent component analysis*. John Wiley & Sons, Hoboken, NJ, USA.

Thorsten Joachims, 1998. Text categorization with support vector machines: Learning with many relevant features. In C. Nédellec and C. Rouveirol, editors, *Machine Learning: ECML-98*, volume 1398 of *Lecture Notes in Computer Science*, pages 137–142. Springer-Verlag, Berlin/Heidelberg, Germany.

Steve Jones and Gordon W. Paynter, 2001. Human evaluation of Kea, an automatic keyphrasing system. In *Proceedings of the First ACM+IEEE Joint Conference on Digital libraries (JCDL)*, pages 148–156, Roanoke, VA, USA. ACM.

Karen Spärck Jones, 1972. A statistical interpretation of term specifity and its application in retrieval. *Journal of Documentation*, 28(1):11–21.

Patrick Juola and R. Harald Baayen, 2005. A controlled-corpus experiment in authorship identification by cross-entropy. *Literary and Linguistic Computing*, 20(Suppl):59–67.

Daniel Jurafsky and James H. Martin, 2009. *Speech and Language Processing*. Pearson Education, Upper Saddle River, NJ, USA, 2nd edition.

Christian Jutten and Jeanny Hérault, 1991. Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal processing*, 24(1):1–10.

Pentti Kanerva, Jan Kristoferson, and Anders Holst, 2000. Random indexing of text samples for latent semantic analysis. In *Proceedings of the 22nd Annual Conference of the Cognitive Science Society (CogSci 2000)*, page 1036, Philadelphia, PA, USA. Lawrence Erlbaum.

Jakub Kanis and Luděk Müller, 2005. Automatic lemmatizer construction with focus on oov words lemmatization. In P. Mautner V. Matoušek and T. Pavelka, editors, *Text, Speech and Dialogue, 8th International Conference, TSD 2005, Karlovy Vary, Czech Republic, September 12-15, 2005*, volume 3658 of *Lecture Notes in Computer Science*, pages 132–139. Springer-Verlag, Berlin/Heidelberg, Germany.

Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman, and Angela Y. Wu, 2002. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):881–892.

Jussi Karlgren and Magnus Sahlgren, 2001. From words to understanding. In *Foundations of Real-World Intelligence*, CSLI Publications, pages 294–308. Center for the Study of Language and Information, Stanford, CA, USA.

Jussi Karlgren, Magnus Sahlgren, Fredrik Olsson, Fredrik Espinoza, and Ola Hamfors, 2012. Usefulness of sentiment analysis. In R. Baeza-Yates, A. P. de Vries, H. Zaragoza, B. B. Cambazoglu, V. Murdock, R. Lempel, and F. Silvestri, editors, *Advances in Information Retrieval, 34th European Conference on IR Research, ECIR 2012, Barcelona, Spain, April 1-5, 2012*, volume 7224 of *Lecture Notes in Computer Science*, pages 426–435. Springer-Verlag, Berlin/Heidelberg, Germany.

Samuel Kaski, Timo Honkela, Krista Lagus, and Teuvo Kohonen, 1998. WEBSOM – self-organizing maps of document collections. *Neurocomputing*, 21(1-3):101–117.

Samuel Kaski, 1998. Dimensionality reduction by random mapping: Fast similarity computation for clustering. In *Proceedings of International Joint Conference on Neural Networks (IJCNN'98)*, volume 1, pages 413–418, Anchorage, AK, USA. IEEE.

Kenneth Katzner, 2002. *The languages of the world*. Routledge, London/New York, 3rd edition.

Su Nam Kim, Timothy Baldwin, and Min-Yen Kan, 2010. Evaluating n-gram based evaluation metrics for automatic keyphrase extraction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling)*, pages 572–580, Beijing, China. Association for Computational Linguistics.

Su Nam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin, 2010. SemEval-2010 Task 5: Automatic keyphrase extraction from scientific articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*, pages 21–26, Uppsala, Sweden. Association for Computational Linguistics.

Su Nam Kim, Lawrence Cavedon, and Timothy Baldwin, 2011. Harvesting domain-specific terms using Wikipedia. In *Proceedings of the Sixteenth Australasian Document Computing Symposium (ADCS)*, Canberra, Australia.

Tibor Kiss and Jan Strunk, 2006. Unsupervised multilingual sentence boundary detection. *Computational Linguistics*, 32(4):485–525.

George R. Kiss, Christine Armstrong, Robert Milroy, and James Piper, 1973. An associative thesaurus of English and its computer analysis. In A. J. Aitkin, R. W. Bailey, and N. Hamilton-Smith, editors, *The Computer and Literary Studies*. Edinburgh University Press, Edinburgh, UK.

Mikaela Klami and Timo Honkela, 2007. Self-organized ordering of terms and documents in NSF awards data. In *Proceedings of the 6th International Workshop on Self-Organizing Maps (WSOM'07)*, Bielefeld, Germany.

Jon M. Kleinberg, 1999. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632.

Philipp Koehn and Josh Schroeder, 2007. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 224–227, Prague, Czech Republic. Association for Computational Linguistic.

Philipp Koehn, Franz Josef Och, and Daniel Marcu, 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT-NAACL)*, pages 48–54, Edmonton, Canada. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst, 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL), Companion Volume: Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Philipp Koehn, 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the 10th Machine Translation Summit*, pages 79–86, Phuket, Thailand. Asia-Pacific Association for Machine Translation.

Philipp Koehn, 2010. *Statistical machine translation*. Cambridge University Press, New York, NY, USA.

Philipp Koehn, 2011. What is a better translation? Reflections of six years of running evaluation campaigns. In *Tralogy 2011*. Online http://homepages.inf.ed.ac.uk/pkoehn/publications/tralogy11.pdf (Accessed 12 April 2012).

Teuvo Kohonen, 2001. *Self-Organizing Maps*. Springer-Verlag, Berlin/Heidelberg/New York.

Pasi Koikkalainen and Erkki Oja, 1990. Self-organizing hierarchical feature maps. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN'90)*, volume 2, pages 279–285, Washington, D.C., USA.

Markus Koskela, Mats Sjöberg, and Jorma Laaksonen, 2009. Improving automatic video retrieval with semantic concept detection. In A.-B. Salberg, J. Y. Hardeberg, and R. Jenssen, editors, *Image Analysis*, volume 5575 of *Lecture Notes in Computer Science*, pages 480–489. Springer-Verlag, Berlin/Heidelberg, Germany.

Zornitsa Kozareva, Eduard Hovy, and Ellen Riloff, 2009. Learning and evaluating the content and structure of a term taxonomy. In *Proceedings of the AAAI Spring Symposium on Learning by Reading and Learning to Read*, pages 50–57, Stanford, USA. The AAAI Press.

Tayfun Kucukyilmaz, B. Barla Cambazoglu, Cevdet Aykanat, and Fazli Can, 2006. Chat mining for gender prediction. In T. Yakhno and E. J. Neuhold, editors, *Advances in Information Systems*, volume 4243 of *Lecture Notes in Computer Science*, pages 274–283. Springer-Verlag, Berlin/Heidelberg, Germany.

Taku Kudo and Yuji Matsumoto, 2001. Chunking with support vector machines. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 192–199, Pittsburgh, PA, USA. Association for Computational Linguistics.

Mikko Kurimo, Sami Virpioja, Ville Turunen, Graeme W. Blackwood, and William Byrne, 2010. Overview and results of Morpho Challenge 2009. In C. Peters, G. M. Di Nunzio, M. Kurimo, T. Mandl, D. Mostefa, A. Peñas, and G. Roda, editors, *Multilingual Information Access Evaluation I. Text Retrieval Experiments*, volume 6241 of *Lecture Notes in Computer Science*, pages 578–597. Springer-Verlag, Berlin/Heidelberg, Germany.

Mikko Kurimo, Sami Virpioja, Ville Turunen, and Krista Lagus, 2010. Morpho Challenge 2005–2010: Evaluations and results. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology (SIGMORPHON)*, pages 87–95, Uppsala, Sweden. Association for Computational Linguistics.

Krista Lagus, Timo Honkela, Samuel Kaski, and Teuvo Kohonen, 1999. WEBSOM for textual data mining. *Artificial Intelligence Review*, 13(5-6):345–364.

George Lakoff, 1987. *Women, fire, and dangerous things: What categories reveal about the mind*. University of Chicago Press, Chicago/London.

George Lakoff, 1993. The contemporary theory of metaphor. In A. Ortony, editor, *Metaphor and Thought*, pages 202–251. Cambridge University Press, Cambridge, UK, 2nd edition.

Thomas K. Landauer and Susan T. Dumais, 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211–240.

Daniel D. Lee and H. Sebastian Seung, 1999. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791.

Stephen C. Levinson, 1983. *Pragmatics*. Cambridge textbooks in linguistics. Cambridge University Press, Cambridge, UK.

David D. Lewis, 1992. An evaluation of phrasal and clustered representations on a text categorization task. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '92)*, pages 37–50, Copenhagen, Denmark. ACM.

Yaoyong Li and John Shawe-Taylor, 2007. Advanced learning algorithms for cross-language patent retrieval and classification. *Information Processing and Management*, 43(5):1183–1199.

Jiexun Li, Rong Zheng, and Hsinchun Chen, 2006. From fingerprint to writeprint. *Communications of the ACM*, 49(4):76–82.

Jefrey Lijffijt, Panagiotis Papapetrou, Kai Puolamäki, and Heikki Mannila, 2011. Analyzing word frequencies in large text corpora using inter-arrival times and bootstrapping. In D. Gunopulos, T. Hofmann, D. Malerba, and M. Vazirgiannis, editors, *Machine Learning and Knowledge Discovery in Databases*, volume 6912 of *Lecture Notes in Computer Science*, pages 341–357. Springer-Verlag, Berlin/Heidelberg, Germany.

Chul Su Lim, Kong Joo Lee, and Gil Chang Kim, 2005. Multiple sets of features for automatic genre classification of Web documents. *Information Processing and Management*, 41(5):1263–1276.

Winston Lin, Roman Yangarber, and Ralph Grishman, 2003. Bootstrapped learning of semantic classes from positive and negative examples. In *Proceedings of ICML-2003 Workshop on The Continuum from Labeled to Unlabeled Data*, Washington D.C., USA.

Krister Lindén and Lauri Carlson, 2010. FinnWordNet – WordNet på finska via översättning (in Swedish). *LexicoNordica – Nordic Journal of Lexicography*, 17:119–140.

Tiina Lindh-Knuutila, Timo Honkela, and Krista Lagus, 2006. Simulating meaning negotiation using observational language games. In P. Vogt, Y. Sugita, E. Tuci, and C. Nehaniv, editors, *Symbol Grounding and Beyond*, volume 4211 of *Lecture Notes in Computer Science*, pages 168–179. Springer-Verlag, Berlin/Heidelberg, Germany.

Marina Litvak, Mark Last, Hen Aizenman, Inbal Gobits, and Abraham Kandel, 2011. DegExt — a language-independent graph-based keyphrase extractor. In E. Mugellini, P. Szczepaniak, M. C. Pettenati, and M. Sokhn, editors, *Advances in Intelligent Web Mastering – 3*, volume 86 of *Advances in Intelligent and Soft Computing*, pages 121–130. Springer-Verlag, Berlin/Heidelberg, Germany.

Xiaoyong Liu, W. Bruce Croft, Paul Oh, and David Hart, 2004. Automatic recognition of reading levels from user queries. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '04)*, pages 548–549, Sheffield, UK. ACM.

Cynthia Macdonald and Graham Macdonald, editors, 1995. *Connectionism: debates on psychological explanation*, volume 2. Blackwell, Oxford, UK; Cambridge, MA, USA.

Ainura Madylova and Şule G. Öğüdücü, 2009. Comparison of similarity measures for clustering Turkish documents. *Intelligent Data Analysis*, 13(5):815–832.

Alexander Maedche, Viktor Pekar, and Steffen Staab, 2003. Ontology learning part one – on discovering taxonomic relations from the Web. In N. Zhong, J. Liu, and Y. Yao, editors, *Web Intelligence*, chapter 14. Springer-Verlag, Berlin/Heidelberg/New York.

Bernardo Magnini and Carlo Strapparava, 2001. Improving user modelling with content-based technique. In M. Bauer, P. J. Gmytrasiewicz, and J. Vassileva, editors, *User Modeling 2001*, volume 2109 of *Lecture Notes in Computer Science*, pages 74–83. Springer-Verlag, Berlin/Heidelberg, Germany.

Mehrdad Mahdavi, Morteza Haghir Chehreghani, Hassan Abolhassani, and Rana Forsati, 2008. Novel meta-heuristic algorithms for clustering web documents. *Applied Mathematics and Computation*, 201(1–2):441–451.

Inderjeet Mani and Eric Bloedorn, 1999. Summarizing similarities and differences among related documents. *Information Retrieval*, 1(1):35–67.

Christopher D. Manning and Hinrich Schütze, 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, MA, USA.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze, 2008. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA.

Andrei A. Markov, 1913. An example of statistical investigation in the text of 'Eugene Onyegin' illustrating coupling of 'tests' in chains. In *Proceedings of the Academy of Sciences, St. Petersburg*, volume 7, pages 153–162.

David Martinez and Eneko Agirre, 2000. One sense per collocation and genre/topic variations. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*

(EMNLP/VLC-2000), pages 207–215, Hong Kong, China. Association for Computational Linguistics.

Yutaka Matsuo and Mitsuru Ishizuka, 2004. Keyword extraction from a single document using word co-occurrence statistical information. *International Journal on Artificial Intelligence Tools*, 13(1):157–169.

Diana McCarthy and Roberto Navigli, 2007. SemEval-2007 task 10: English lexical substitution task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval)*, pages 48–53, Prague, Czech Republic. Association for Computational Linguistics.

Diana McCarthy, 2002. Lexical substitution as a task for WSD evaluation. In *Proceedings of the SIGLEX/SENSEVAL Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*, pages 109–115, Philadelphia, PA, USA. Association for Computational Linguistics.

Peter McCullagh and John A. Nelder, 1990. *Generalized Linear Models*. Chapman & Hall, New York, NY, USA.

G. Harry McLaughlin, 1969. SMOG grading – a new readability formula. *Journal of Reading*, 12(8):639–646.

Rada Mihalcea and Paul Tarau, 2004. TextRank: Bringing order into texts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Rada Mihalcea and Paul Tarau, 2005. A language independent algorithm for single and multiple document summarization. In *Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP)*, pages 19–24, Jeju Island, Korea.

Rada Mihalcea, Ravi Sinha, and Diana McCarthy, 2010. SemEval-2010 Task 2: Cross-lingual lexical substitution. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*, pages 9–14, Uppsala, Sweden. Association for Computational Linguistics.

Rada Mihalcea, 2004. Co-training and self-training for word sense disambiguation. In *Proceedings of HLT-NAACL 2004 Workshop: the Eighth Conference on Computational Natural Language Learning (CoNLL-2004)*, pages 33–40, Boston, MA, USA. Association for Computational Linguistics.

George A. Miller, 1995. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41.

Jeff Mitchell and Mirella Lapata, 2008. Vector-based models of semantic composition. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT)*, pages 236–244, Columbus, OH, USA. Association for Computational Linguistics.

Pabitra Mitra, C. A. Murthy, and Sankar K. Pal, 2002. Unsupervised feature selection using feature similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):301–312.

Arto Mustajoki, 2008. Modelling of (mis)communication. In *Prikladna lingvistika ta ligvistitshni tehnologii: Megaling-2007*, pages 250–267.

Peter Nabende, 2011. Mining transliterations from Wikipedia using dynamic Bayesian networks. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP)*, pages 385–391, Hissar, Bulgaria. RANLP 2011 Organising Committee.

Nikolaos Nanas, Manolis Vavalis, and Elias Houstis, 2010. Personalised news and scientific literature aggregation. *Information Processing and Management*, 46(3):268–283.

National Library of Medicine, 1960. *Medical subject headings: main headings, subheadings, and cross references used in the Index Medicus and the National Library of Medicine Catalog.* U.S. Department of Health, Education, and Welfare, Washington, D.C., USA, 1st edition.

Douglas L. Nelson, Cathy L. McEvoy, and Thomas A. Schreiber. The University of South Florida word association, rhyme, and word fragment norms. Online http://web.usf.edu/FreeAssociation/ (Accessed 7 Oct 2010), 1998. University of South Florida, Tampa, FL, USA.

Graham Neubig, Taro Watanabe, Eiichiro Sumita, Shinsuke Mori, and Tatsuya Kawahara, 2011. An unsupervised model for joint phrase alignment and extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL: HLT)*, pages 632–641, Portland, OR, USA. Association for Computational Linguistics.

Mark E. J. Newman, 2005. Power laws, Pareto distributions and Zipf's law. *Contemporary Physics*, 46(5):323–351.

Vincent Ng and Claire Cardie, 2003. Weakly supervised natural language learning without redundant views. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT-NAACL)*, pages 94–101, Edmonton, Canada. Association for Computational Linguistics.

Vincent Ng, 2008. Unsupervised models for coreference resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 640–649, Honolulu, HI, USA. Association for Computational Linguistics.

Thuy Dung Nguyen and Minh-Thang Luong, 2010. WINGNUS: Keyphrase extraction utilizing document logical structure. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*, pages 166–169, Uppsala, Sweden. Association for Computational Linguistics.

Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülşen Eryiğit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi, 2007. MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Franz J. Och and Hermann Ney, 2000. Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics (ACL)*, pages 440–447, Hong Kong, China. Association for Computational Linguistics.

Charles K. Ogden and Ivor A. Richards, 1923. *The meaning of meaning*. Harcourt Brace Jovanovich, New York, NY, USA.

Erkki Oja, 1982. A simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology*, 15:267–273.

Roberto Ortiz, David Pinto, Mireya Tovar, and Héctor Jiménez-Salazar, 2010. BUAP: An unsupervised approach to automatic keyphrase extraction from scientific articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*, pages 174–177, Uppsala, Sweden. Association for Computational Linguistics.

Patrick Pantel and Dekang Lin, 2001. A statistical corpus-based term extractor. In E. Stroulia and S. Matwin, editors, *Advances in Artificial Intelligence*, volume 2056 of *Lecture Notes in Computer Science*, pages 36–46. Springer-Verlag, Berlin/Heidelberg, Germany.

Claude Pasquier, 2010. Single document keyphrase extraction using sentence clustering and latent Dirichlet allocation. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*, pages 154–157, Uppsala, Sweden. Association for Computational Linguistics.

Ted Pedersen, Amruta Purandare, and Anagha Kulkarni, 2005. Name discrimination by clustering similar contexts. In A. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, volume 3406 of *Lecture Notes in Computer Science*, pages 226–237. Springer-Verlag, Berlin/Heidelberg, Germany.

Ted Pedersen, Anagha Kulkarni, Roxana Angheluta, Zornitsa Kozareva, and Thamar Solorio, 2006. An unsupervised language independent method of name discrimination using second order co-occurrence features. In A. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, volume 3878 of *Lecture Notes in Computer Science*, pages 208–222. Springer-Verlag, Berlin/Heidelberg, Germany.

Charles Sanders Peirce, 1931–1958. *Collected papers of Charles Sanders Peirce*, volume I-VIII. Harvard University Press, Cambridge, MA, USA.

Saverio Perugini, 2010. Personalization by website transformation: Theory and practice. *Information Processing and Management*, 46(3):284–294.

Sarah E. Petersen and Mari Ostendorf, 2009. A machine learning approach to reading level assessment. *Computer Speech and Language*, 23(1):89–106.

Emanuele Pianta and Sara Tonelli, 2010. KX: A Flexible System for Keyphrase eXtraction. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*, pages 170–173, Uppsala, Sweden. Association for Computational Linguistics.

Frans Plank and Elena Filimonova, 2000. The universals archive: a brief introduction for prospective users. *Sprachtypologie und Universalienforschung*, 53:109–123.

Simone Paolo Ponzetto and Michael Strübe, 2007. Deriving a large scale taxonomy from Wikipedia. In *Proceedings of the Twenty-Second Conference on Artificial Intelligence (AAAI-07)*, pages 1440–1445, Vancouver, Canada. The AAAI Press.

Sameer Pradhan, Kadri Hacioglu, Valerie Krugler, Wayne Ward, James H. Martin, and Daniel Jurafsky, 2005. Support vector learning for semantic argument classification. *Machine Learning*, 60(1):11–39.

Geoffrey K. Pullum, 1977. Word order universals and grammatical relations. In P. Cole and J. M. Sadock, editors, *Grammatical Relations*, volume 8 of *Syntax and Semantics*. Academic Press, New York, NY, USA.

Amruta Purandare and Ted Pedersen, 2004. Word sense discrimination by clustering contexts in vector and similarity spaces. In *Proceedings of HLT-NAACL 2004 Workshop: the Eighth Conference on Computational Natural Language Learning (CoNLL-2004)*, pages 41–48, Boston, MA, USA. Association for Computational Linguistics.

Amruta Purandare, 2003. Discriminating among word senses using McQuitty's similarity analysis. In *Proceedings of the HLT-NAACL 2003 Student Research Workshop*, volume 3, pages 19–24, Edmonton, Canada. Association for Computational Linguistics.

Silvia Quarteroni and Suresh Manandhar, 2006. User modelling for adaptive question answering and information retrieval. In *Proceedings of Proceedings of the Nineteenth International Florida Artificial Intelligence Research Society Conference (FLAIRS)*, pages 776–781, Melbourne Beach, FL, USA. AAAI Press.

Silvia Quarteroni and Suresh Manandhar, 2007. User modelling for personalized question answering. In R. Basili and M. T. Pazienza, editors, *AI*IA 2007: Artificial Intelligence and Human-Oriented Computing*, volume 4733 of *Lecture Notes in Computer Science*, pages 386–397. Springer-Verlag, Berlin/Heidelberg, Germany.

Dragomir R. Radev, Simone Teufel, Horacio Saggion, Wai Lam, John Blitzer, Hong Qi, Arda Çelebi, Danyu Liu, and Elliott Drabek, 2003. Evaluation challenges in large-scale document summarization. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics (ACL)*, pages 375–382, Sapporo, Japan. Association for Computational Linguistics.

Reinhard Rapp and Michael Zock, 2010. Automatic dictionary expansion using non-parallel corpora. In W. Seidel A. Fink, B. Lausen and A. Ultsch, editors, *Advances in Data Analysis, Data Handling and Business Intelligence*, Studies in Classification, Data Analysis, and Knowledge Organization, pages 317–325. Springer-Verlag, Berlin/Heidelberg, Germany.

Reinhard Rapp, 1995. Identifying word translations in non-parallel texts. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 320–322, Cambridge, MA, USA. Association for Computational Linguistics.

Reinhard Rapp, 1999. Automatic identification of word translations from unrelated English and German corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 519–526, College Park, MD, USA. Association for Computational Linguistics.

Reinhard Rapp, 2002. The computation of word associations: Comparing syntagmatic and paradigmatic approaches. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING'02)*, pages 1–7, Taipei, Taiwan. Association for Computational Linguistics.

Paul Rayson, Damon Berridge, and Brian Francis, 2004. Extending the Cochran rule for the comparison of word frequencies between corpora. In *7th International Conference on Statistical Analysis of Textual Data (JADT 2004)*, pages 926–936.

Ehud Reiter and Somayajulu Sripada, 2004. Contextual influences on near-synonym choice. In A. Belz, R. Evans, and P. Piwek, editors, *Natural Language Generation*, volume 3123 of *Lecture Notes in Computer Science*, pages 161–170. Springer-Verlag, Berlin/Heidelberg, Germany.

Korin Richmond, Andrew Smith, and Einat Amitay, 1997. Detecting subject boundaries within text: A language independent statistical approach. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing (EMNLP '97)*, pages 47–54, Providence, RI, USA.

Helge Ritter and Teuvo Kohonen, 1989. Self-organizing semantic maps. *Biological Cybernetics*, 61:241–254.

Peter Roget, 1911. *Thesaurus of English Words and Phrases*. Longmans, Green and Co., London, UK.

Eleanor Rosch, 1978. Principles of categorization. In E. Rosch and B. B. Lloyd, editors, *Cognition and categorization*, pages 27–48. Lawrence Erlbaum, Hillsdale, NJ, USA.

Gerda Ruge, 1992. Experiments on linguistically-based term associations. *Information Processing and Management*, 28(3):317–332.

Rudolph J. Rummel, 1970. *Applied Factor Analysis*. Northwestern University Press, Evanston, IL, USA.

Tuukka Ruotsalo, Lora Aroyo, and Guus Schreiber, 2009. Knowledge-based linguistic annotation of digital cultural heritage collections. *IEEE Intelligent Systems*, 24(2):64–75.

Markus Sadeniemi, Kimmo Kettunen, Tiina Lindh-Knuutila, and Timo Honkela, 2008. Complexity of European Union languages: A comparative approach. *Journal of Quantitative Linguistics*, 15(2):185–211.

John I. Saeed, 1997. *Semantics*. Blackwell Publishers Ltd, Oxford, UK.

Magnus Sahlgren, 2006. Towards pertinent evaluation methodologies for word-space models. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy. European Language Resources Association.

Magnus Sahlgren. 2006. *The Word-Space Model*. PhD thesis, Department of Linguistics, Stockholm University, Stockholm, Sweden.

Magnus Sahlgren and Jussi Karlgren, 2005. Automatic bilingual lexicon acquisition using random indexing of parallel corpora. *Natural Language Engineering*, 11(03):327–341.

Magnus Sahlgren and Jussi Karlgren, 2009. Terminology mining in social media. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM'09)*, pages 405–414, Hong Kong, China. ACM.

Magnus Sahlgren, 2001. Vector-based semantic analysis: representing word meanings based on random labels. In *Semantic Knowledge Acquisition and Categorisation Workshop at European Summer School in Logic, Language and Information (ESSLLI XIII)*, Helsinki, Finland. Kluwer.

Jarkko Salojärvi, Kai Puolamäki, and Samuel Kaski, 2004. Relevance feedback from eye movements for proactive information retrieval. In *Workshop on Processing Sensory Information for Proactive Systems (PSIPS 2004)*, Oulu, Finland.

Gerard Salton and Christopher Buckley, 1988. Term-weighting approaches in automatic text retrieval. *Information Prosessing and Management*, 24(5):513–523.

Gerard Salton, Anita Wong, and Chung-Shu Yang, 1975. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

Gerard Salton, editor, 1971. *The SMART System — Experiments in Automatic Document Processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

David Sánchez and Antonio Moreno, 2005. Web-scale taxonomy learning. In *Proceedings of Workshop on Extending and Learning Lexical Ontologies using Machine Learning, ICML05*, pages 53–60, Bonn, Germany.

Mark Sanderson and Bruce Croft, 1999. Deriving concept hierarchies from text. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on*

Research and Development in Information Retrieval (SIGIR '99), pages 206–213, Berkeley, CA, USA. ACM.

Roger C. Schank and Robert P. Abelson, 1975. Scripts, plans, and knowledge. In *Advance papers of the Fourth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 151–157, Tblisi, Georgia, USSR.

Adam Schenker, Mark Last, Horst Bunke, and Abraham Kandel, 2003. Comparison of distance measures for graph-based clustering of documents. In E. Hancock and M. Vento, editors, *Graph Based Representations in Pattern Recognition*, volume 2726 of *Lecture Notes in Computer Science*, pages 202–213. Springer-Verlag, Berlin/Heidelberg, Germany.

Patrick Schone and Daniel Jurafsky, 2001. Language-independent induction of part of speech class labels using only language universals. In *Proceedings of the IJCAI-2001 Workshop "Text Learning: Beyond Supervision"*, pages 53–60, Seattle, WA, USA.

Hinrich Schütze and Jan O. Pedersen, 1997. A cooccurrence-based thesaurus and two applications to information retrieval. *Information Processing and Management*, 33(3):307–318.

Hinrich Schütze, 1992. Dimensions of meaning. In *Proceedings of the 1992 ACM/IEEE Conference on Supercomputing (SC 1992)*, pages 787–796, Minneapolis, MN, USA. IEEE Computer Society.

Hinrich Schütze, 1997. *Ambiguity Resolution in Language Learning: Computational and Cognitive Models*. Number 71 in CSLI Lecture Notes. Center for the Study of Language and Information, Stanford, CA, USA.

Mike Scott, 2001. Mapping key words to problem and solution. In M. Scott and G. Thompson, editors, *Patterns of Text: in honour of Michael Hoey*, pages 109–128. John Benjamins B.V.

Fabrizio Sebastiani, 2002. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47.

R. J. Senter and E. A. Smith. 1967. Automated readability index. Technical Report AMRL-TR 66-22, Wright-Patterson AFB, OH, USA.

Claude E. Shannon, 1948. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423,623–656.

Serge Sharoff, 2010. Analysing similarities and differences between corpora. In T. Erjavec and J. Žganec Gros, editors, *Proceedings of the 13th International Multiconference Information Society (IS 2010)*, volume C, Language Technologies, pages 5–11, Ljubljana, Slovenia.

John Shawe-Taylor and Nello Cristianini, 2004. *Kernel Methods for Pattern Analysis*. Cambridge University Press, Cambridge, UK.

Edward E. Smith and Douglas L. Medin, 1981. *Categories and Concepts*. Harvard University Press, Cambridge, MA, USA; London, UK.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng, 2006. Semantic taxonomy induction from heterogenous evidence. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics (Coling-ACL)*, pages 801–808, Sydney, Australia. Association for Computational Linguistics.

Thamar Solorio, Manuel Pérez-Coutiño, Manuel Montes-y-Gómez, Luis Villaseñor-Pineda, and Aurelio López-López, 2004. A language independent method for question classification. In *Proceedings of the 20th International Conference on Computational Linguistics (Coling)*, pages 1374–1380, Geneva, Switzerland. Association for Computational Linguistics.

Luc Steels, 2008. The symbol grounding problem has been solved, so what's next? In M. de Vega, editor, *Symbols and Embodiment: Debates on Meaning and Cognition*, chapter 12, pages 223–244. Oxford University Press, Oxford, UK.

Ralf Steinberger, Flavio Fuart, Erik van der Goot, Clive Best, Peter von Etter, and Roman Yangarber, 2008. Text mining from the Web for medical intelligence. In F. Fogelman-Soulié, D. Perrotta, J. Piskorski, and R. Steinberger, editors, *Mining Massive Data Sets for Security*, pages 295–310. IOS Press, The Netherlands.

Alexander Strehl, Joydeep Ghosh, and Raymond Mooney, 2000. Impact of similarity measures on web-page clustering. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-00)*, pages 58–64, Austin, TX, USA. AAAI/The MIT Press.

Richard S. Sutton and Andrew G. Barto, 1998. *Reinforcement Learning: An Introduction*. The MIT Press, Cambridge, MA, USA; London, UK.

Mariarosaria Taddeo and Luciano Floridi, 2005. Solving the symbol grounding problem: a critical review of fifteen years of research. *Journal of Experimental & Theoretical Artificial Intelligence*, 17(4):419–445.

Pasi Tapanainen and Timo Järvinen, 1997. A non-projective dependency parser. In *Proceedings of the 5th Conference on Applied Natural Language Processing (ANLP)*, pages 64–71, Washinton, D.C., USA.

Jaime Teevan, Susan T. Dumais, and Eric Horvitz, 2005. Personalizing search via automated analysis of interests and activities. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '05)*, pages 449–456, Salvador, Brazil. ACM.

Aristomenis Thanopoulos, Nikos Fakotakis, and George Kokkinakis, 2002. Comparative evaluation of collocation extraction metrics. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC 2002)*, pages 620–625, Las Palmas, Spain. European Language Resources Association.

Sergios Theodoridis and Konstantinos Koutroumbas, 2008. *Pattern Recognition*. Academic Press, New York, NY, USA, 4th edition.

Rob Thomson and Tamar Murachver, 2001. Predicting gender from electronic discourse. *British Journal of Social Psychology*, 40:193–208.

Erik F. Tjong Kim Sang and Fien De Meulder, 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In W. Daelemans and M. Osborne, editors, *Proceedings of the Seventh Conference on Computational Natural Language Learning (CoNLL-2003)*, pages 142–147, Edmonton, Canada. Association for Computational Linguistics.

Erik F. Tjong Kim Sang, 2002. Introduction to the CoNLL-2002 shared task: language-independent named entity recognition. In D. Roth and A. van den Bosch, editors, *Proceedings of the Sixth Workshop on Computational Language Learning (CoNLL-2002)*, pages 155–158, Taipei, Taiwan. Association for Computational Linguistics.

Peter P. Toma, 1977. Systran as a multilingual machine translation system. In *Proceedings of the Third European Congress on Information Systems and Networks, Overcoming the language barrier*, pages 569–581.

Peter D. Turney, 2000. Learning algorithms for keyphrase extraction. *Information Retrieval*, 2(4):303–336.

Peter D. Turney, 2001. Mining the Web for synonyms: PMI-IR versus LSA on TOEFL. In L. De Raedt and P. A. Flach, editors, *Machine Learning: ECML 2001*, volume 2167 of *Lecture Notes in Computer Science*, pages 491–502. Springer-Verlag, Berlin/Heidelberg, Germany.

Peter D. Turney, 2005. Measuring semantic similarity by latent relational analysis. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)*, pages 1136–1141, Edinburgh, UK. Morgan Kaufmann Publishers.

Vladimir N. Vapnik, 1995. *The nature of statistical learning theory*. Information Science and Statistics. Springer-Verlag, New York, NY, USA.

Tommi Vatanen, Jaakko J. Väyrynen, and Sami Virpioja, 2010. Language identification of short text segments with n-gram models. In N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, pages 3423–3430, Valletta, Malta. European Language Resources Association.

Harri Veivo and Tomi Huttunen, 1999. *Semiotiikka (in Finnish)*. Edita, Helsinki, Finland.

Paola Velardi, Alessandro Cucchiarelli, and Michaël Pétit, 2007. A taxonomy learning method and its application to characterize a scientific web community. *IEEE Transactions on Knowledge and Data Engineering*, 19:180–191.

Manisha Verma and Vasudeva Varma, 2011. Applying key phrase extraction to aid invalidity search. In *Proceedings of the 13th International Conference on*

Artificial Intelligence and Law (ICAIL), pages 249–255, Pittsburgh, PA, USA. ACM.

Kees Versteegh, 1997. *The Arabic language*. Edinburgh University Press, Edinburgh, UK.

Alexei Vinokourov, John Shawe-Taylor, and Nello Cristianini, 2003. Inferring a semantic representation of text via cross-language correlation analysis. *Advances in Neural Information Processing Systems*, 15:1497–1504.

Sami Virpioja, Jaakko J. Väyrynen, Mathias Creutz, and Markus Sadeniemi, 2007. Morphology-aware statistical machine translation based on morphs induced in an unsupervised manner. In *Proceedings of the Machine Translation Summit XI*, pages 491–498, Copenhagen, Denmark.

Sami Virpioja, Minna Lehtonen, Annika Hultén, Riitta Salmelin, and Krista Lagus, 2011. Predicting reaction times in word recognition by unsupervised learning of morphology. In T. Honkela, W. Duch, M. Girolami, and S. Kaski, editors, *Artificial Neural Networks and Machine Learning – ICANN 2011*, volume 6791 of *Lecture Notes in Computer Science*, pages 275–282. Springer–Verlag, Berlin/Heidelberg, Germany.

Sami Virpioja, Ville T. Turunen, Sebastian Spiegler, Oskar Kohonen, and Mikko Kurimo, 2011. Empirical comparison of evaluation methods for unsupervised learning of morphology. *Traitement Automatique des Langues*, 52(2):45–90.

Jorge Vivaldi and Horacio Rodríguez, 2010. Finding domain terms using Wikipedia. In N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, pages 386–393, Valletta, Malta. European Language Resources Association.

Adam Vogel and Dan Jurafsky, 2010. Learning to follow navigational directions. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 806–814, Uppsala, Sweden. Association for Computational Linguistics.

Paul Vogt, 2003. Anchoring of semiotic symbols. *Robotics and Autonomous Systems*, 43:109–120.

Ulrike von Luxburg, 2007. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416.

Ellen M. Voorhees, 1994. Query expansion using lexical-semantic relations. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '94)*, pages 61–69, Dublin, Ireland. Springer-Verlag, New York, NY, USA.

Annika Wærn, 2004. User involvement in automatic filtering: An experimental study. *User Modeling and User-Adapted Interaction*, 14:201–237.

Tong Wang and Graeme Hirst, 2010. Near-synonym lexical choice in latent semantic space. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling)*, pages 1182–1190, Beijing, China. Association for Computational Linguistics.

Xing Wei and W. Bruce Croft, 2006. LDA-based document models for ad-hoc retrieval. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '06)*, pages 178–185, Seattle, WA, USA. ACM.

Casey Whitelaw, Ben Hutchinson, Grace Y Chung, and Gerard Ellis, 2009. Using the Web for language independent spellchecking and autocorrection. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 890–899, Singapore. Association for Computational Linguistics.

Casey Whitelaw and Jon Patrick, 2003. Named entity recognition using a character-based probabilistic approach. In W. Daelemans and M. Osborne, editors, *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL-2003)*, pages 196–199, Edmonton, Canada. Association for Computational Linguistics.

Edward W. D. Whittaker, Julien Hamonic, Dong Yang, Tor Klingberg, and Sadaoki Furui, 2006. Monolingual Web-based factoid question answering in Chinese, Swedish, English and Japanese. In *Proceedings of the Workshop on Multilingual Question Answering*, pages 45–52, Trento, Italy. Association for Computational Linguistics.

Anna Wierzbicka, 1999. *Emotions across Languages and Cultures: Diversity and Universals*. Cambridge University Press, Cambridge, UK.

Wilson Y. Wong. 2009. *Learning Lightweight Ontologies from Text across Different Domains using the Web as Background Knowledge*. PhD thesis, The University of Western Australia, Perth, Australia.

Jia-Long Wu and Alice M. Agogino, 2003. Automating keyphrase extraction with multi-objective genetic algorithms. In *Proceedings of the Hawaii International Conference on System Science (HICSS)*, Big Island, HI, USA.

Fei Wu and Daniel S. Weld, 2010. Open information extraction using Wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 118–127, Uppsala, Sweden. Association for Computational Linguistics.

Ying Wu, Thomas S. Huang, and Kentaro Toyama, 2001. Self-supervised learning for object recognition based on kernel discriminant-EM algorithm. In *Proceedings of the Eighth IEEE International Conference on Computer Vision (ICCV'01)*, volume 1, pages 275–280, Vancouver, Canada. IEEE.

Roman Yangarber, 2004. User-oriented evaluation in information extraction. In *The Proceedings of the Workshop on User-Oriented Evaluation of Knowledge Discovery Systems*, Lisbon, Portugal.

David Yarowsky, 1993. One sense per collocation. In *Proceedings of the Workshop on Human Language Technology (HLT'93)*, pages 266–271, Plainsboro, NJ, USA. Association for Computational Linguistics.

David Yarowsky, 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting on Association for Computational Linguistics (ACL)*, pages 189–196, Cambridge, MA, USA. Association for Computational Linguistics.

Deniz Yuret and Mehmet Ali Yatbaz, 2010. The noisy channel model for unsupervised word sense disambiguation. *Computational Linguistics*, 36(1):111–127.

Taras Zagibalov and John Carroll, 2008. Almost-unsupervised cross-language opinion analysis at NTCIR-7. In *Proceedings of NTCIR-7 Workshop*, pages 204–210, Tokyo, Japan.

Torsten Zesch and Iryna Gurevych, 2009. Wisdom of crowds versus wisdom of linguists — measuring the semantic relatedness of words. *Natural Language Engineering*, 16(1):25–59.

Ying Zhao and George Karypis, 2004. Empirical and theoretical comparisons of selected criterion functions for document clustering. *Machine Learning*, 55(3):311–331.

Ying Zhao and George Karypis, 2005. Hierarchical clustering algorithms for document datasets. *Data Mining and Knowledge Discovery*, 10:141–168.

Shaojun Zhao, 2004. Named entity recognition in biomedical texts using an HMM model. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA '04)*, pages 84–87, Geneva, Switzerland. Association for Computational Linguistics.

Rong Zheng, Jiexun Li, Hsinchun Chen, and Zan Huang, 2006. A framework for authorship identification of online messages: Writing-style features and classification techniques. *Journal of the American Society for Information Science and Technology*, 57(3):378–393.

Shi Zhong and Joydeep Ghosh, 2005. Generative model-based document clustering: a comparative study. *Knowledge and Information Systems*, 8(3):374–384.

Xiaojin Zhu and Andrew B. Goldberg, 2009. *Introduction to semi-supervised learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, USA.

George Kingsley Zipf, 1949. *Human behavior and the principle of least effort*. Addison-Wesley, Cambridge, MA, USA.

Pierre Zweigenbaum, Pierre Jacquemart, Natalia Grabar, and Benoît Habert, 2001. Building a text corpus for representing the variety of medical language. In *Proceedings of the 10th World Congress on Medical Informatics (medinfo2001)*, pages 290–294, London, UK.

DISSERTATIONS IN INFORMATION AND COMPUTER SCIENCE

Aalto-DD136/2011	Brumley, Billy Bob Covert Timing Channels, Caching, and Cryptography. 2011.
Aalto-DD11/2012	Vuokko, Niko Testing the Significance of Patterns with Complex Null Hypotheses. 2012.
Aalto-DD19/2012	Reunanen, Juha Overfitting in Feature Selection: Pitfalls and Solutions. 2012.
Aalto-DD33/2012	Caldas, José Graphical Models for Biclustering and Information Retrieval in Gene Expression Data. 2012.
Aalto-DD45/2012	Viitaniemi, Ville Visual Category Detection: an Experimental Perspective. 2012.
Aalto-DD51/2012	Hanhijärvi, Sami Multiple Hypothesis Testing in Data Mining. 2012.
Aalto-DD56/2012	Ramkumar, Pavan Advances in Modeling and Characterization of Human Neuromagnetic Oscillations. 2012.
Aalto-DD97/2012	Turunen, Ville T. Morph-Based Speech Retrieval: Indexing Methods and Evaluations of Unsupervised Morphological Analysis. 2012.
Aalto-DD115/2012	Vierinen, Juha On statistical theory of radar measurements. 2012.
Aalto-DD117/2012	Huopaniemi, Ilkka Multivariate Multi-Way Modelling of Multiple High-Dimensional Data Sources. 2012.

This dissertation was typed with pdfLaTeX, using typographical look-and-feel `classicthesis`, modified by André Miede and further modified by the author.

Social media and the whole Internet have enabled an enormous growth in the amount of unstructured text. This text is a mixture of multiple languages, domains, writing styles and both standard and non-standard usage of languages. This kind of free-form text cannot be successfully processed with traditional natural language processing tools based on rule-based parsers, dictionaries and predefined lists of acceptable terms. In this dissertation, language-independent methods for automatic natural language processing are developed and discussed. Language independence relies on unsupervised machine learning paradigms and collects the information of the language usage from large unstructured text collections. Language-independent methods can be used for many tasks in any language or domain, such as information retrieval in search engines, analyzing and summarizing the contents of large document collections and as the basis of text processing systems which accept subjective variations in language use.

Aalto-DD 137/2012

ISBN 978-952-60-4833-8
ISBN 978-952-60-4834-5 (pdf)
ISSN-L 1799-4934
ISSN 1799-4934
ISSN 1799-4942 (pdf)

Aalto University
School of Science
Department of Information and Computer Science
www.aalto.fi

BUSINESS + ECONOMY

ART + DESIGN + ARCHITECTURE

SCIENCE + TECHNOLOGY

CROSSOVER

DOCTORAL DISSERTATIONS