Available online at www.sciencedirect.com

ScienceDirect

Computer Speech and Language 35 (2016) 163–184

COMPUTER SPEECH AND LANGUAGE

www.elsevier.com/locate/csl

State of the art in statistical methods for language and speech processing☆

Jerome R. Bellegarda[a,*], Christof Monz[b]

[a] *Apple Inc., One Infinite Loop, Cupertino, CA 95014, USA*
[b] *Informatics Institute, University of Amsterdam, Science Park 904, 1098 XH Amsterdam, The Netherlands*

Received 14 November 2014; received in revised form 21 May 2015; accepted 3 July 2015
Available online 14 July 2015

Abstract

Recent years have seen rapid growth in the deployment of statistical methods for computational language and speech processing. The current popularity of such methods can be traced to the convergence of several factors, including the increasing amount of data now accessible, sustained advances in computing power and storage capabilities, and ongoing improvements in machine learning algorithms. The purpose of this contribution is to review the state of the art in both areas, point out the top trends in statistical modelling across a wide range of problems, and identify their most salient characteristics. The paper concludes with some prognostications regarding the likely impact on the field going forward.

Keywords: Statistical methods for language processing; Speech processing

1. Introduction

Statistical methods can be thought of as a way to leverage information primarily extracted from available raw data rather than derived from *a priori* expert knowledge. Over the past decade, such approaches have enjoyed rapidly increasing popularity in the field of computational language and speech processing. A wide spectrum of machine learning techniques have now been deployed to address the full complement of problems to be solved, from speech recognition and natural language modelling to information retrieval and text summarisation. This enthusiasm for statistical methods is the consequence of two main developments: (i) the explosion in the amount of data newly accessible, largely due to new social behaviours, societal transformations, as well as the vast spread of software systems and (ii) a steady reduction in the cost of computing and storage resources, which makes it possible to process increasingly large quantities of data with a reasonable amount of time and money.

☆ This paper has been recommended for acceptance by Shrikanth Narayanan.
* Corresponding author. Tel.: +1 408 974 7647.
E-mail addresses: jerome@apple.com (J.R. Bellegarda), c.monz@uva.nl (C. Monz).

http://dx.doi.org/10.1016/j.csl.2015.07.001
0885-2308/© 2015 Elsevier Ltd. All rights reserved.

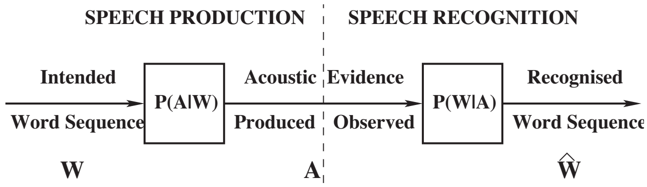


Fig. 1. Speech processing seen as information transmission over a noisy channel. The "Transmission Channel" is symbolised by the dashed line separating speech production from speech recognition.

1.1. Background

Arguably, the field of language and speech processing is inherently well suited for the kind of statistical methods now commonly adopted for data analytics under the "Big Data" paradigm. As early as the mid-1970s, speech processing began to be seen as an instance of information transmission over a noisy channel (Bahl et al., 1983). This view led to the well-known framework depicted in Fig. 1, where W refers to a word or sequence of words to be produced, A to the acoustic realisation of the resulting textual data, and $\hat{W}$ to the recognised sequence recovered from the observed evidence.

The framework of Fig. 1 comprises blocks labelled with parameters of the form $P(\cdot|\cdot)$, reflecting a noisy process characterised by a statistical model. The output sequence $\hat{W}$ then satisfies:

$$\hat{W} = arg\max_{W \in \mathcal{S}} P(W|A), \tag{1}$$

where the maximisation is done over all possible candidate word sequences in some feasible set $\mathcal{S}$. Using Bayes' rule and the fact that the maximisation is independent of the observation likelihood $P(A)$, (1) can also be written as:

$$\hat{W} = arg\max_{W \in \mathcal{S}} P(A|W)P(W), \tag{2}$$

which exposes two fundamental statistical models: the acoustic model $P(A|W)$, which characterises speech production, and the language model $P(W)$, which expresses the *a priori* probability of generating a particular sequence W in the language. These two models have formed the basis of automatic speech recognition (ASR) for the past three decades (Rabiner et al., 1996).

Interestingly, the same view can also encompass higher levels of language processing. For example, the framework of Fig. 2 corresponds to the "personal assistant" paradigm associated with products likeApple Siri (Apple Inc., 2011), Google Now (Google Inc., 2012), or Microsoft Cortana (Microsoft Corp., 2014). In that scenario, the user wants a particular problem solved, which is formulated as intent I. This intent is conveyed through a particular query sequence W, itself leading to an acoustic realisation as before. The recognised sequence $\hat{W}$ then elicits one of several possible actions, aimed at fulfilling the original intent I. Note that the blocks $P(W|I)$ and $P(I|W)$ play similar roles at the language level as the blocks $P(A|W)$ and $P(W|A)$ play at the speech level.

Given that both language and speech processing lend themselves particularly well to the application of statistical models, it is of interest to examine recent trends in each of the two research communities, along with their most salient characteristics.

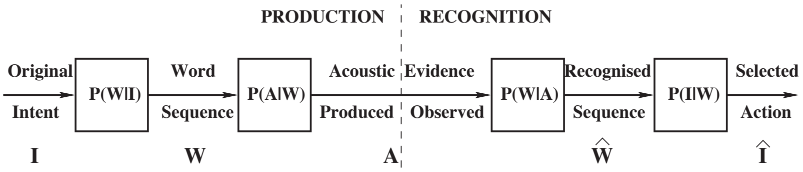


Fig. 2. An example of integrated language and speech processing: personal assistance seen as information transmission over a noisy channel.

1.2. Recent trends in language processing

Computational language processing covers a wide spectrum of research areas, addressing diverse end-to-end tasks such as document summarisation (Hovy and Lin, 1997; Knight and Marcu, 2001; Zajic et al., 2005; Kastner and Monz, 2009) and machine translation (Lopez, 2008; Koehn, 2009), as well as more linguistically motivated challenges, such as syntactic parsing (Collins, 1999; Bikel, 2004) and part-of-speech tagging (Schmid, 1994; Brill, 1995). In the following we primarily focus on trends that are visible across several of these sub-areas of language processing. Note that deep learning has also been applied to some of these tasks, but will be discussed in Section 3 in the specific context of acoustic and language modelling for speech recognition.

Historically speaking, language processing has its roots in artificial intelligence (AI), and like most early approaches in AI, research in language processing tended to be mostly symbolic. Here, we use the term ‘symbolic’ to refer to approaches that are mostly rule-based, where rules tend to be hand-crafted and rule-formation is guided by linguistic intuition. The early 90s marked a watershed moment for language processing (Marcus, 1994), where statistical methods were successfully applied to several language processing tasks, such as syntactic parsing and part-of-speech tagging. In the following two decades, statistical approaches have become the main-stream. Large and annotated data sets allowed for rules to be learned automatically by applying machine learning methods.

At this present day, a number of trends can be observed, all of which relate to this shift from symbolic to statistical approaches. Defining accurate semantic, i.e. meaning, representations is one of the most challenging tasks in language processing. Early approaches relied on first-order logic as a formal semantic representation (Montague, 1970). It was soon realised that first-order logic was not expressive enough to capture all the subtleties of human language resulting in a plethora of extensions and alternatives to first-order logic (Chierchia, 1992; Kamp and Reyle, 1993; Muskens, 1996; Dekker, 2000). A recent trend in computational semantics is to move away from explicit, formal knowledge representations and instead represent the meaning of a phrase or sentence using *distributional semantics*. In distributional semantics, the meaning of a word is computed based on the context in which a word tends to occur. The meaning of larger constituents, such as a phrase or sentence, is then computed based on the principle of compositionality combining the meaning representations of the smaller elements making up that constituent.

A consequence of moving from symbolic to statistical approaches is the increasing importance of data. Most statistical approaches are data-driven. On the one hand, this means that in order to address a certain language processing task the availability of sufficient training data becomes a vital pre-condition. At the same time, the availability of certain data can also inspire researchers to address new tasks. Over the last decade social media created vast amounts of *user-generated content* such as weblogs and microblogs (e.g. tweets). User-generated data is not only of societal importance, as evidenced by the staggering numbers of users, but also presents researchers with a tremendous array of challenges and opportunities. Unlike traditional data used to train statistical language processing models, user-generated content contains significantly higher levels of noise, ranging from spelling and grammatical mistakes, to semantic vagueness: What exactly is the meaning of a ‘like’ on Facebook?

The last trend we would like to discuss here is the increasing importance of *evaluation* in language processing and the usage of crowd-sourcing to provide human assessments as well as to generate annotated benchmarks. The increase in evaluation campaigns over the last few years further encourages researchers to compare their approaches in an objective manner. Many of these more recent evaluation campaigns are run by members of the research community themselves. As manual annotation is labour intensive, some evaluation campaigns have recently turned to crowd-sourcing platforms such Amazon’s Mechanical Turk or CrowdFlower, which offers immense opportunities to obtain manual assessments for a large variety of tasks, but at the same time also requires novel methods and practices to reduce and remove noise due to non-expert annotations.

1.3. Recent trends in speech processing

In speech processing, *deep learning* has emerged as the most prominent trend over the past few years. Most systems used to implement the acoustic model $P(A|W)$ of (2) by relying on hidden Markov models (HMMs) to deal with the temporal variability of speech, and Gaussian mixture models (GMMs) to determine how well each state of each HMM fits a frame (or a short window of frames) of coefficients representing the acoustic input. An alternative way to evaluate the fit is to use a neural network which takes several frames of coefficients as input and produces posterior probabilities over HMM states as output. Deep neural networks (DNNs), which comprise many fully connected hidden layers as

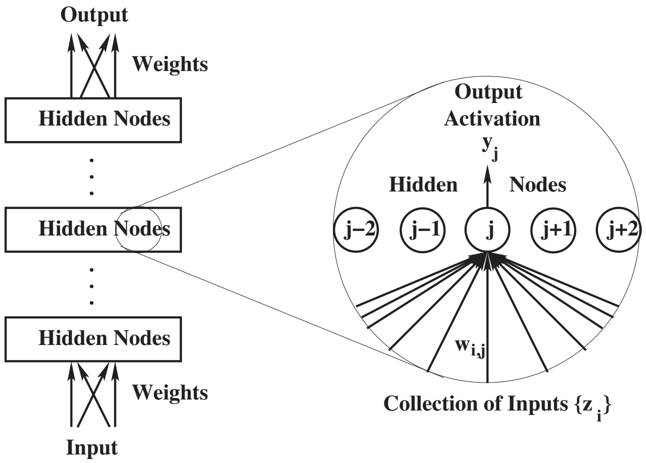


Fig. 3. A generic deep neural network with typical bottom-up, directed connections. The symbol ↑↗↖↑ represents full connections between layers.

depicted in Fig. 3, have been shown to outperform GMMs on a variety of speech recognition benchmarks, sometimes by a wide margin (Hinton et al., 2012).

A similar shift has been happening in language modelling for speech recognition. Most systems used to implement the language model $P(W)$ of (2) via n-grams, sub-sequences of n consecutive words in a corpus of text. An alternative way is to use a (deep) neural network which takes (suitable representations of) words as input and produces as output probabilities conditioned on the entire history available. Particularly promising are models based on recurrent neural networks (RNNs), which in theory allow for an infinite history (Mikolov et al., 2010).

Another major trend revolves around the idea of *promoting sparsity*. Generally speaking, the goal of statistical modelling is to establish a generalisation from a set of observed data such that accurate inference (classification, decision, recognition) can be made about the data yet to be observed (unseen, or test, data). When the test sample is used to inform the construction of the statistical model, it becomes possible to judiciously select a subset of exemplars from the training data to build a local model specifically for every test sample. In the machine learning community, this is referred to as lazy (instance-based or memory-based) learning. In the speech community, this is referred to as sparse representations or exemplar-based modelling, since the model is built from a few relevant training examples for each test sample. Recent advances in computing power and storage and improvements in machine learning algorithms have made exemplar-based techniques successful in increasingly complex speech tasks (Sainath et al., 2012).

The final trend considered in this paper concerns *robust feature extraction and knowledge integration*. Speech is characterised by specific spectro-temporal patterns. For the past quarter of a century the standard approach to extract these patterns has been of two main flavours: Mel-frequency cepstral coefficients (MFCCs) (Furui, 1981) or perceptual linear predictive coefficients (PLPs) (Hermansky, 1990). Both types are computed from the raw waveform and their first- and second-order temporal differences. This nonadaptive but highly engineered preprocessing of the waveform is designed to serve two purposes: (i) discard the large amount of information in waveforms deemed irrelevant for discrimination and (ii) express the remaining information in a form that facilitates discrimination with GMM-HMMs. More recently, other feature extraction strategies have shown promise, such as “tandem” or bottleneck features generatedusing DNNs, and integrated features computed by adding convolutional layers to a traditional DNN (Sainath et al., 2013). Some efforts have also been expanded to leverage a new understanding of the influence of cortical regions mediating categorical speech perception (Mesgarani et al., 2014). Finally, a significant amount of work has gone to the integration of those heterogeneous knowledge sources, along with other structural information potentially available, in order to increase the overall robustness of the deployed systems (Zweig and Nguyen, 2010).

1.4. Outline of the paper

This contribution focuses on the recent developments in computational language and speech processing identified above, striving to give a working overview of each and put them into perspective with regard to the state of the art.

While necessarily non-exhaustive given space constraints, this overview aims at finding common ground amid the wide spectrum of statistical methods currently in use in the two research communities, in the hope of fostering tighter collaboration and more cross-fertilisation. The material is organised as follows. The next section discusses three major trends in language processing, covering three diverse aspects of recent developments: distributional semantics, user-generated content, and evaluation. Section 3 does the same for the speech processing trends: deep learning, sparsity promotion, and knowledge integration. Finally, Section 4 concludes the paper with a summary of the material covered, and some prognostications regarding the likely impact on the field going forward.

2. Language processing trends

In this section, we discuss three recent trends in computational language processing, covering three different aspects of current developments in this broad area. First, we address distributional semantics as a statistical approach in computational semantics. Then, we review recent trends in moving away from the traditionally-used training data towards phenomena encountered in user-generated content. Finally, we discuss current developments in evaluation, and in particular the benefits and challenges of evaluation campaigns.

2.1. Distributional semantics

Although distributional semantics marks a significant departure from earlier approaches to semantic representation in natural language processing, many of its core ideas can be traced back to the early stages of modern of natural language philosophy and linguistic theory.

2.1.1. Context as meaning

Most semantic approaches that map words directly to terms in a formal knowledge representation language, such as $[\![\text{car}]\!] = \lambda x\,.\,car(x)$, where $[\![\cdot]\!]$ is a function mapping natural language expressions, in this case ‘car’, to formal representations, and $\lambda x\,.\,car(x)$ represents the set of all cars.

The core intuition underlying distributional semantics on the other hand is that words have the same meaning if they occur in the same contexts. Therefore, the meaning of a word can be formalised by a representation of its contexts. Even though most people will not know what a ‘mumma’ is, when reading the sentence ‘We all enjoyed a traditional glass of mumma while unwrapping our gifts’, it will become clear that it is a drink enjoyed on festive occasions.

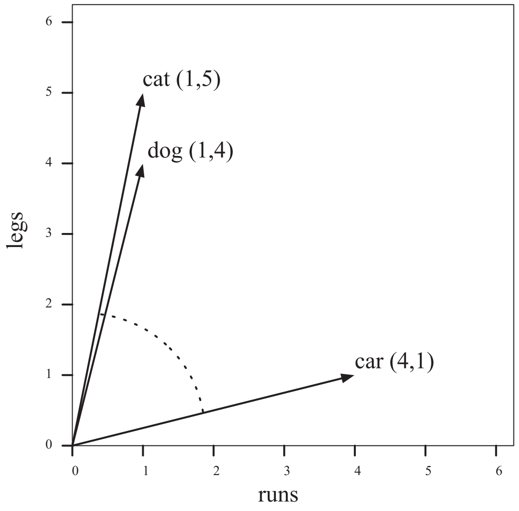
If meanings of words can be represented by their contexts, the next question is how to represent contexts. Currently, distributional semantics chooses a fairly simple format, where a context of a word w is represented as a vector v_w storing the frequencies (including zero-frequencies) of all words co-occurring with w in a corpus (Turney and Pantel, 2010).

Fig. 4 shows the context vectors for six words. One can see that some vectors have a similar *distribution* of frequencies, indicating that they are similar in meaning. Representing contexts as vector also allows one to make use of standard algebraic operations such as the dot product to compute the cosine similarity between two vectors (Fig. 5). Refinements on this basic strategy have led to techniques such as latent semantic mapping (LSM) in text indexing and retrieval (Bellegarda, 2005), which results in a parsimonious continuous parameter description of words and contexts via a projection onto the most information-bearing lower-dimensional subspace (Bellegarda, 2008).

word	context vector					
	leash	walk	run	owner	pet	bark
dog	3	5	2	5	3	2
cat	0	3	3	2	3	0
lion	0	3	2	0	1	0
light	0	0	0	0	0	0
bark	1	0	0	2	1	0
car	0	0	1	3	0	0

Fig. 4. Context vectors for the words ‘dog’, ‘cat’, ’lion’, ‘light’, ’bark’, and ‘car’. Each element in the vector represents a co-occurrence count with the respective word.

word	context vector	
	runs	legs
dog	1	4
cat	1	5
car	4	1



(a) (b)

Fig. 5. (a) Shows the (simplified) context vectors for the word ‘dog’, ‘cat’, and ‘car’. (b) Displays the same information in a 2-dimensional space where the angle between vectors can be used as a distance measure.

There are various ways to define the scope of a context, ranging from a sliding window covering *n* tokens to the left and right, to whole sentences and even to whole documents. In addition, there are a number of ways to define the elements that constitute contexts. Above, each dimension of the context vector corresponds to a word in the vocabulary, that is the actual surface, i.e. fully inflected, form. On the other hand, higher levels of abstraction can be achieved by using stemmed forms, part-of-speech tags, or grammatical relationships.

Distributional semantics also shares some commonalities with word embeddings commonly used in probabilistic neural network language modeling (Bengio et al., 2003) and statistical approaches to learning word representations such as the skip-gram approach within `word2vec` (Mikolov et al., 2013) and GloVe (Pennington et al., 2014). Whereas distributional semantics uses high-dimensional vector representations, where each dimension corresponds to a word in the vocabulary, word embedding approaches map words to a set of latent variables (Yih et al., 2012; Chang et al., 2013). While context vectors in distribution semantics tend to be high-dimensional and sparse, word embeddings use lower dimensional and dense vectors, where the value of each dimension is a numerical real value. Fig. 6 shows the architecture of the skip-gram model.

In the skip-gram model, the task is to predict words $w(t-2), \ldots, w(t+2)$ that occur in the context of word $w(t)$. Word embeddings are captured by the projection layer in Fig. 6, where the values of the hidden projection layer are optimized based on the accuracy of the model in prediction the correct context words. Word embeddings can not only capture the similarity of between related words, but also allow for simple arithmetic operations, allowing one to infer analogous relationships between words such as $e(king) - e(man) + e(woman) = e(queen)$, where $e(\cdot)$ represents a word embedding (Mikolov et al., 2013). The resulting word embeddings of the skip-gram model are fairly generous, and have been shown beneficial for a variety of natural language processing tasks, ranging from sentiment detection (Kim, 2014) to Chinese word segmentation (Liu et al., 2014).

2.1.2. *Compositionality of meaning*

So far we have focused on representing the meaning of individual words. Of course, one could simply extend this to representing the meaning of larger units such as phrases or even sentences by simply computing the distributions over the contexts surrounding those larger units. The disadvantage is that this can easily lead to data sparsity issues. Fortunately, this can be addressed by relying on the linguistic principle of compositionality which states that the meaning of a complex expression is determined by the meanings of its constituent expressions and the rules used to combine them (Partee, 1984). Representing the meaning of constituents as vectors could allow us to use simple algebraic operations to combine meaning representations of smaller constituents to compute an approximate meaning

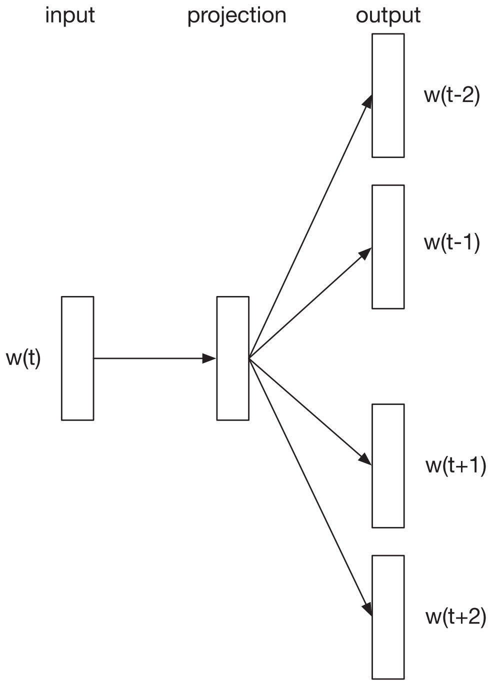


Fig. 6. Network architecture of `word2vec`'s skip-gram model.

representations of larger constituents. Unfortunately, different algebraic operations suffer from different shortcomings. For instance, vector addition and multiplication are order-insensitive, ignoring the order in which words are uttered or written (Mitchell and Lapata, 2008). Recent approaches aim to address this issue by using mixture models using a weighted sum of additive and multiplicative models (Mitchell and Lapata, 2010), but the issue of how to estimate these weights in general remains. More recently, Socher et al. (2013) introduced an approach based on recursive neural networks that allows for a more principled way to account for compositionality in the context of sentiment detection.

While there are still a number of important issues that remain to be solved, distributional semantics provides a first significant step towards a general computational approach to meaning that does not depend on an elaborate formal representation language.

2.2. User-generated content

Unlike the first trend, which was mostly methodological in nature, the second trend directly concerns data. Since the emergence of data-driven approaches in language processing, data plays a central role. Data is used to train statistical models and also to evaluate their performance. In the early days of statistical language processing a small number of training corpora were used to perform a limited number of language processing tasks, such as the Brown corpus for part-of-speech tagging, and the UPenn Treebank corpus (Marcus et al., 1993) for syntactic parsing. While the number of corpora and the tasks they were intended for grew over the years, they had a number of things in common: Firstly, the texts themselves tended to be written by professional authors, such as journalists, undergoing editorial control. Secondly, the corpora were carefully annotated with linguistic information, such as part-of-speech tags, word-senses, or syntactic structures, adhering to comprehensive annotation rules and guidelines (Bies et al., 1995), which in same cases also highlighted the difficulty of defining a 'correct' annotation (Levy and Manning, 2003). Thirdly, they tended to be rather limited in size due to the human effort and cost of annotation.

With the rise of the Internet, researchers started to explore its usefulness for language processing tasks. Much of the data crawled from the Internet does not undergo editorial control, and therefore has a higher proportion of typographical, grammatical, and factual mistakes. On the other hand, the large amount of available data could compensate for these shortcomings. For instance, Lapata and Keller (2005) show that word n-gram statistics obtained from crawled web data can be used successfully for a number of language processing tasks. Brants et al. (2007) show that n-gram statistics based on very large web data sets allow for much simpler n-gram smoothing methods than the commonly-used Kneser–Ney smoothing method (Chen and Goodman, 1999) while yielding similar performance in a machine translation task.

Even though web-crawled documents lack the linguistic annotations needed to train statistical models for traditional language processing tasks such as parsing, they do offer other types of annotations such as hyperlinks and semi-structured document markup. This in turn can be exploited to automatically find relationships between documents, such as determining whether two documents are parallel texts, i.e. one is a translation of the other (Resnik and Smith, 2003).

The rise of Web 2.0 has meant that users are enabled and encouraged to provide content themselves, leading to web-forums, weblogs, commenting sections on articles, product reviews, social network websites such as Facebook, collaborative authoring websites such as Wikipedia, and micro-blogs such as Twitter and Weibo. Each of these sources provides certain types of annotations leading to a plethora of ways in which these annotations can be analysed and exploited for numerous tasks. At the same time, they also present a significant challenge for existing language processing approaches and models. Whereas most language processing models are trained on clean and editorially checked data, user-generated data is much noisier, contains more errors, and is much more geared towards an informal style of writing. This difference also highlights one of the potential vulnerabilities of statistical approaches: the assumption that training and test data come from the same distribution. The effect of this difference can for example be seen in machine translation. Since 2006, NIST's MT Evaluation Campaign (MT-Eval) contains test data that are partially drawn from newswire sources and partially from Internet forums and weblogs, while the training data originates mostly from newswire and United Nation sources. In the latest MT-Eval campaign, the difference in translation quality between translations of newswire documents and translations of weblogs and Internet forums was about 10 BLEU points on average for Arabic-to-English and Chinese-to-English translation. This shows that the state of the art in machine translation is far away from achieving robust translation quality that generalises well beyond the phenomena observed during training.

Recently, some efforts were made to bias the estimates of the general training data towards the test data consisting of user-generated content. For instance, Jehl et al. (2012) improve machine translation of tweets by using document retrieval methods to identify related tweets in both source and target language and adding them to the training data. As micro-blog posts, such as tweets, are rarely posted as direct translations of other tweets, it is important to be able to automatically identify tweets that are translations of each other. Ling et al. (2013) show that automatic identification of tweet translations, including fragment translation, and using these tweets to augment the training data, can lead to substantial improvements in translation quality.

Similar to tweets, also SMS text messages, pose a problem to current language processing methods. The annual Workshop on Machine Translation (WMT), recently devoted a shared task to this problem (Callison-Burch et al., 2011) consisting of text messages that were sent during the January 2010 earthquake in Haiti to an emergency response service. Participants were faced with a number of problems ranging from 'text speak' to the lack of punctuation (Eidelman et al., 2011).

The short nature of tweets and SMS text messages turns even the most basic of language processing tasks into a challenge. For instance, language identification, which is a prerequisite for all further analysis tasks, such as part-of-speech tagging or morphological analysis, is far from trivial for very short segments (Carter et al., 2013). Similarly, part-of-speech tagging of micro-blogs yields lower accuracy when trained on commonly used annotated data from the newswire domain. Owoputi et al. (2013) show that unsupervised word clustering methods can improve tagging accuracies for tweets and IRC texts.

As the higher levels of noise present in user-generated data make it more difficult to combine this data with existing clean data, or match models trained on clean data with unseen user-generated data, it is a common strategy to pre-process user-generated content. Pre-processing can include mapping spelling mistakes to their correct form (Stymne, 2011) and orthographic normalisation (Ling et al., 2013; Hassan and Menezes, 2013; Eisenstein, 2013; Yang and Eisenstein, 2013).

Since most user-generated content documents tend to be rather short, which applies in particular to micro-blogs, it is difficult to interpret them in isolation and it is often beneficial to contextualise them in order to facilitate further analysis. In many cases it is possible to link micro-blog messages to full documents such as news articles (Guo et al., 2013). Alternatively, one can group or cluster different micro-blog messages together according to hidden properties, for example representing demographic characteristics (Bergsma and Van Durme, 2013).

Just as it can be beneficial to contextualise micro-blog messages, the reverse also holds. For example, tweets can provide additional information on news items, and general events. As tweets express a user's own opinion or stance on certain topics they can be analysed for various purposes, including voting intentions (Lampos et al., 2013), and even for predicting outcomes of National Football League games (Sinha et al., 2013).

Many of the challenges of analysing user-generated content with existing language processing techniques are to some extent due to the lack of manually annotated training data covering user-generated content. On the other hand, user-generated content contains other types of annotations that make it useful for many novel types of analysis. This includes hash tags and reposts in tweets, hyperlinks and reply chains in weblogs, and likes and shared links in social networks. Most of these annotations are in the form of labels which are explicitly provided by the users. In addition, also more implicit types of annotations can be analysed, such as link click behaviour (Joachims, 2002), search query logs (Gao et al., 2006), and search query reformulations (Riezler and Liu, 2010). A particularly rich form of implicit annotation can be found on collaborative authoring websites, such as Wikipedia. Here, several users create and revise content, with all revisions being stored by the system. Analysing the revisions allows for categorising the type of revision, e.g. factual vs. fluency changes (Bronner and Monz, 2012; Daxenberger and Gurevych, 2013), or developing spelling correction systems.

Wikipedia is also commonly used to populate ontologies. A shortcoming of traditionally used taxonomies, such as WordNet (Fellbaum, 1998), is their inherent incompleteness, in particular at the leaf level. Ponzetto and Strube (2007) analyse the internal link structure between different Wikipedia articles, where the links have been created by users, to induce semantic relationships. Nastase and Strube (2013) expand this line of research by exploiting the fact that many Wikipedia pages, or topics, are present in multiple languages to induce multi-lingual concepts.

2.3. *Crowd-sourced evaluation*

The last trend we want to discuss here concerns the methodologically important aspect of assessing and quantifying the quality of language processing approaches. Evaluation has always played a central role for statistical approaches to language processing. Parts of the language processing community also adapted the concept of evaluation campaigns early on, such as NIST's Text Retrieval Conference (TREC) for information retrieval and question answering and NIST's Message Understanding Conference (MUC) for information extraction, and more recently NIST's Text Analysis Conference (TAC, Surdeanu, 2013) for ontology population.

Over the last decade, the area of language processing saw an increased proliferation of evaluation campaigns, sometimes also known as shared tasks which are organized by the research community itself. This is partly due to the increase of tasks addressed by language processing research, but also by the desire to have objective benchmarks to measure the advances in addressing a specific problem. Nowadays, evaluation campaigns that are run by the research community itself include the Workshop for Statistical Machine Translation (WMT, Bojar et al., 2014), and the International Workshop for Spoken Language Translation (IWSLT, Cettolo et al., 2013) for machine translation, the Recognizing Textual Entailment (RTE) challenge (Bentivogli et al., 2011) for semantic inference, and the Conference for Natural Language Learning (CoNLL, Ng et al., 2013) for various language processing tasks, such as error correction and semantic labelling.

Another important outcome of evaluation campaigns, in addition to providing an overview of the state of the art in a specific research area, is the release of the test data. In some cases, releasing test data can be a bit more involved. For instance, Twitter data cannot be released as such, due to the fact that the copyright rests with the individual user who created a tweet. Each Twitter user has the option to retract a tweet from the public domain. Therefore Twitter data is mostly released as a collection of identifiers linking to tweets, which may or may not have been retracted (Ling et al., 2013).

Since evaluation benchmarks play an important role in language processing research, and in the decision of which research solutions are worth pursuing further, it is crucial that they are carefully selected, prepared, and annotated. One obvious desideratum for test sets is that they are annotated by several human judges and that the judges are in agreement. For many language processing tasks this is not easy to achieve. For instance, Callison-Burch et al. (2010) report just fair to moderate inter-annotator agreement when judges have to rank a few translations of the same source sentence coming from different machine translation systems.

Unfortunately, manual annotation is costly, both in terms of time and money. This issue becomes particularly relevant for evaluation campaigns run by the research community itself where resources dedicated to annotation efforts are scarce. To overcome this obstacle some evaluation campaigns have recently turned to crowd-sourcing platforms such Amazon's Mechanical Turk or CrowdFlower. Crowd-sourcing allows researchers to hire lay Internet users, also known as 'Turkers', to annotate a predefined task for a very moderate payment (Negri and Mehdad, 2010). Analysing the results when using lay annotations shows some promise, but also some problems. Providing appropriate linguistic annotations can be a non-trivial task, sometimes requiring expert knowledge in natural language processing, which most Turkers will lack (Callison-Burch and Dredze, 2010).

Even tasks, where linguistic knowledge is not explicitly required, such as judging the quality of different machine translation outputs, the inter- and intra-annotator agreements between crow-sourced Mechanical Turk users is markedly lower than the corresponding agreements between experts (Callison-Burch et al., 2010). On the other hand, the higher levels of noise found in crowd-sourced annotations become less significant when using more coarse-grained evaluation criteria. For example, Bloodgood and Callison-Burch (2010) compare several machine translation systems against crowd-sourced annotations and NIST's official annotations created by professional translators. While the absolute scores differ somewhat, the ranking of the various systems remains stable. The agreement between the crowd-sourced and official annotations can be further increased by post-editing the crowd-sourced annotations, showing the need for some manual inspection and quality control of crowd-sourced annotations.

Lower inter-annotator agreements between Turkers than experts could also be attributed to the complexity of the task and the extent to which it requires NLP expertise or intuition. Heilman and Smith (2010) report relatively high inter-annotator agreements across Turkers and experts for a simpler task, where annotators judge the quality of automatically generated reading comprehension questions on a five-point scale.

In addition to post-editing or filtering the crowd-sourced annotations, Lawson et al. (2010) proposed a bonus strategy that encourages high-quality Turkers to undertake larger annotations efforts by increasing their monetary reward relative to other Turkers. The challenge is to automatically measure quality of a Turker. In general, there are two strategies: (1) measure accuracy on control data annotated by an expert and (2) measure agreement between different Turkers.

Annotation quality can also be increased by adjusting and simplifying the task. Negri et al. (2011) used crowd-sourcing to annotate a data set with cross-lingual textual entailment relationships. As a whole, this is a rather complex task for non-experts, requiring bilingual expertise as well as a good understanding of the notion of entailment. To reduce the cognitive load for the non-expert annotators, this task is split into a sequence of easier sub-tasks, such as paraphrasing and assessing grammaticality only. In some cases, sub-tasks can also be organized in a way such that Turkers work on each others annotations. For example, Marge et al. (2010) task several Turkers to independently generate speech transcripts for the same source. Disagreements between the transcripts are automatically detected, and other Turkers are then asked to resolve those cases only.

Williams et al. (2011) found that even just splitting the annotation tasks into smaller chunks increased overall accuracy in a speech transcription task. At the same time, they observed that breaking an annotation effort into smaller tasks, where each task tends to result in a lower monetary reward for the annotator, took longer for the overall annotation to be completed.

Besides the language processing areas mentioned above, including machine translation speech recognition and textual entailment, crowd-sourced evaluation sets and annotations have been used in several other language processing tasks such as named entity recognition (Finin et al., 2010), semantic relationship identification (Gormley et al., 2010), and word sense disambiguation (Hong and Baker, 2011).

3. Speech processing trends

In this section, we first address deep learning in acoustic and language modelling for speech recognition, then review sparse representations and exemplar-based processing, and finally discuss robust feature extraction and the integration of heterogeneous knowledge sources.

3.1. Deep learning

The use of neural networks in acoustic modelling is not new: two decades ago, researchers achieved some success using networks with a single layer of non-linear hidden units to predict HMM states from windows of acoustic coefficients (Bourlard and Morgan, 1993). Referring back to Fig. 3 with only one layer of hidden nodes, denote the collection of input values by $\{z_i\}$, and the collection of weights pertinent to the *j*th hidden node by $\{w_{i,j}\}$. Each hidden node j in the network may be viewed as a correlation filter mapping its total input to a scalar output activation of the form:

$$y_j = \sigma(b_j + \sum_i w_{i,j}\, z_i), \tag{3}$$

where $\sigma(\cdot)$ is an appropriate non-linear function (typically the logistic function or the closely related hyperbolic tangent), and b_j is the intended bias of the unit. Fundamentally, the unit j “fires” if the correlation between the inputs and their weights exceeds a threshold $-b_j$. Thus, collectively the nodes in the hidden layer may be viewed as detectors of basic patterns. Output units then convert these basic patterns into class probabilities $\{p_j\}$ by using the “softmax” non-linearity:

$$p_j = \frac{\exp(y_j)}{\sum_k \exp(y_k)}, \tag{4}$$

where k is an index ranging over all classes considered, in this case the collection of HMM states.

In the early 1990s, however, neither the hardware nor the learning algorithms were adequate for training neural networks with many hidden layers on large amounts of data, and the benefits of using neural networks with a single hidden layer were not sufficiently large to seriously challenge GMMs. As a result, until a few years ago the main practical contribution of neural networks was to provide extra features in “tandem” (Zhu et al., 2005) or bottleneck systems (Grezl et al., 2007).

3.1.1. Deep learning for acoustic modelling

More recently, advances in both machine learning algorithms and computer hardware have led to more efficient methods for training DNNs that contain many layers of non-linear hidden units and a very large output layer. The large output layer is required to accommodate the large number of HMM states that arise when each phone is modelled by a number of different “triphone” HMMs that take into account the phones on either side. Even when many of the states of these triphone HMMs are tied together, there can be thousands of tied states.

Historically, one key development has been *generative pre-training*. The idea is to initialise each layer of the DNN individually, by learning one layer of feature detectors at a time, with the states of the feature detectors in one layer acting as the data for training the next layer. Not only does this strategy offer a much better starting point for standard backpropagation training, it also significantly reduces overfitting (Larochelle et al., 2007). In practice, each layer of the DNN is treated as a restricted Boltzmann machine (RBM), and an undirected generative model learns the joint probability of the observable and latent variables. This is done as depicted in Fig. 7. First, a Gaussian-Bernoulli RBM (GRBM) is trained to model a window of frames of real-valued acoustic coefficients, leading to the matrix of weights $\mathbf{W}_1$. Then the states of the binary hidden units of the GRBM are used as data for training an RBM, leading to the matrix of weights $\mathbf{W}_2$. This is repeated to create as many hidden layers as desired. Each time, the inferred states of the hidden units are used as data for training another RBM that learns to model the significant dependencies between

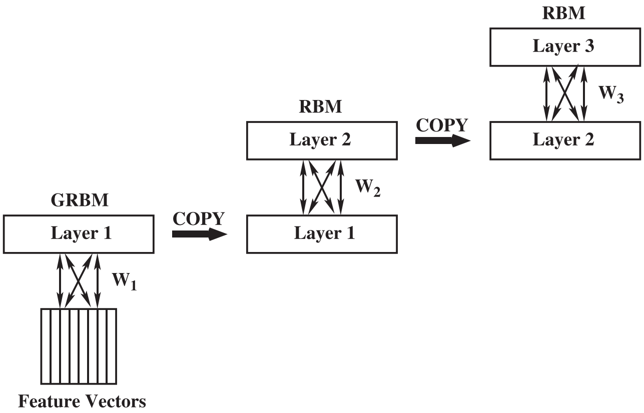


Fig. 7. RBM training of individual layers. Double-ended arrows indicate undirected connections, which facilitate inference of the states of latent variables from the observed data.

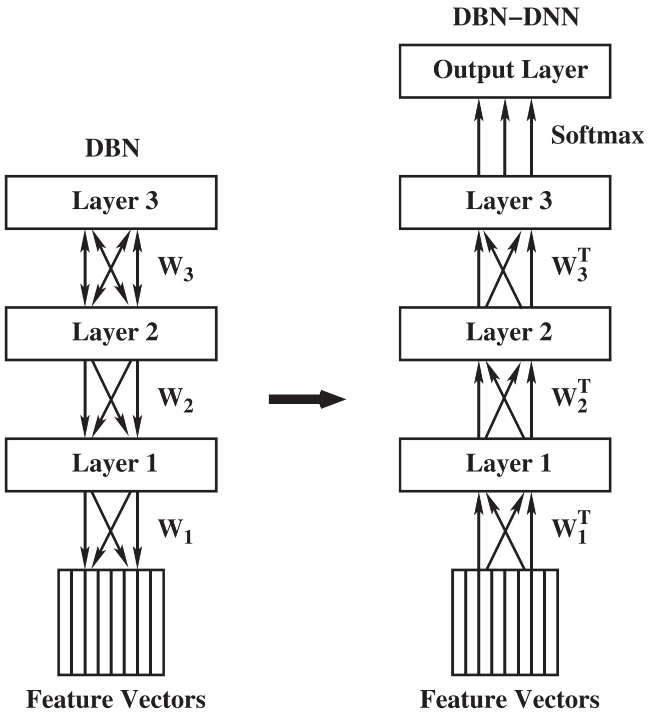


Fig. 8. Converting a stack of RBMs to a DBN hybrid generative model (Left) and then to a pre-trained DNN (Right). The symbol ^{T} denotes transposition.

the hidden units of the current RBM. The outcome is thus a stack of individually trained RBMs, where each layer of non-linear feature detectors uncovers progressively more complex statistical structure in the data.

As depicted on the left hand side of Fig. 8, this stack of RBMs is then converted to a single, multilayer generative model called a deep belief network (DBN) (Hinton et al., 2006). Even though each RBM is an undirected model, the DBN formed by the whole stack is a hybrid generative model whose top two layers are undirected (they constitute the final RBM in the stack) but whose lower layers have top-down, directed connections. Thus the DBN results from replacing the undirected connections of the lower level RBMs by top-down, directed connections. The key property of a DBN that distinguishes it from other multilayer, directed, non-linear generative models is that it is possible to infer the states of the layers of hidden units in a single forward pass. This inference is not exactly correct but is fairly accurate (Hinton et al., 2006). So after learning a DBN by training a stack of RBMs, we can simply use the generative weights in the reverse direction as a way of initialising all the feature-detecting layers of a deterministic feedforward DNN. This leads to the pre-trained DBN-DNN depicted on the right hand side of Fig. 8, where the connections are now bottom-up. It then suffices to add a "softmax" output layer that contains one unit for each possible state of each HMM, and to train the entire network discriminatively to predict the HMM state corresponding to the central frame of the input window in a forced alignment. The final network conforms exactly to the configuration of Fig. 3.

This new learning method turned out to be a critical catalyst, leading several different research groups to show that DNNs can outperform GMMs at acoustic modelling for speech recognition on a variety of data sets, including large data sets with large vocabularies (Hinton et al., 2012). Since then, it has turned out that as long as "enough" training data is available, generative pre-training is not strictly necessary, and equally good results can be achieved without it (Seide et al., 2011; Yu et al., 2013). As a result, considerable attention is now being spent on efficient training implementations allowing the processing of a large quantity of data in a reasonable amount of time. Two main avenues are being pursued: distributing (CPU) computations on a massive scale (as in the DistBelief model (Dean et al., 2012; Heigold et al., 2014)), and relying on various forms of GPU optimization (Seide et al., 2014, 2014).

Another important area for DNN-based ASR is speaker adaptation. Various methods have been proposed, including replacing the input layer of the DNN by a small neural network to learn a speaker code and a feature transform (Abdel-Hamid and Jiang, 2013), using a speaker i-vector (Dehak et al., 2011) as an additional input to the feature layer of the DNN (Gupta et al., 2014), augmenting the speaker-independent DNN with additional layers (Yao et al., 2012), adapting the activation function (Siniscalchi et al., 2012), changing the target distribution in backpropagation (Yu et al., 2013), and/or using a speaker-adapted feature space (Seide et al., 2011; Saon et al., 2013). As an in-depth analysis of such methods is beyond the scope of this paper, the reader is invited to consult the references above for details.

Note that, while DNNs are typically trained to classify individual frames, speech recognition is inherently a sequence classification problem (Heigold et al., 2014). Training therefore benefits from using sequence-discriminative criteria like maximum mutual information (MMI) (Bahl et al., 1986), boosted MMI (BMMI) (Povey et al., 2008), minimum phone error (MPE) (Povey, 2003) or state-level minimum Bayes risk (sMBR) (Kingsbury et al., 2012). All of these criteria have been shown to lower word error rates by roughly 10% relative, on the average (Veselý et al., 2013). Another way to account for sequential characteristics is via recurrent network architectures. We will treat that subject below when reviewing ASR language modelling, since such architectures have a longer history in that context.

3.1.2. Deep learning for ASR language modelling

Just like in acoustic modelling, the use of neural networks in language modelling is not new. For example, the so-called continuous-space language model relies on a neural network to simultaneously project word indices onto a continuous space and estimate probabilities on that space (Schwenk and Gauvain, 2002). Since the resulting probability functions are smooth functions of the word representation, better generalisation to unknown events can be achieved (Bengio et al., 2003). This is still fundamentally an n-gram approach, but the LM probabilities are interpolated for any possible context of length $n-1$ instead of backing-off to shorter contexts. This approach was successfully used in large-vocabulary continuous speech recognition and in phrase-based statistical translation systems (Schwenk, 2007).

More recently, researchers have experimented with neural network language models using *recurrent connections* (Mikolov et al., 2010). In its simplest form, an RNN has only one hidden layer, but, as illustrated in Fig. 9, this (so-called context or state) layer gets fed back to the input at the next time step. In this way, information can cycle inside the network for an arbitrarily long time, meaning that in principle an infinite history can be taken into account. RNNs have been shown to substantially outperform state of the art backoff n-gram models, even when trained on a smaller amount of data (Wu et al., 2012). However, they tend to struggle when it comes to capturing truly long context information, largely due to the vanishing gradient problem (Hochreiter and Schmidhuber, 1997). This issue can be addressed with more complex implementations, such as Long Short-Term Memory (LSTM) networks (Sundermeyer et al., 2012).

An LSTM node contains gates that determine when the input is significant enough to remember, when the memory cell should continue to remember or forget the value, and when it should output the value (Hochreiter and Schmidhuber, 1997). Thus LSTM implementations are able to handle generic time series with variable time lags between important events, making them attractive to perform sequence learning. Training proceeds as with regular RNNs, for example via iterative gradient descent methods like backpropagation through time (Sundermeyer et al., 2012). Within each LSTM node, however, when error values are back-propagated from the output, the error effectively becomes trapped in the memory cell. It is thus able to continuously feed back to each of the gates until the network is trained to cut the value.

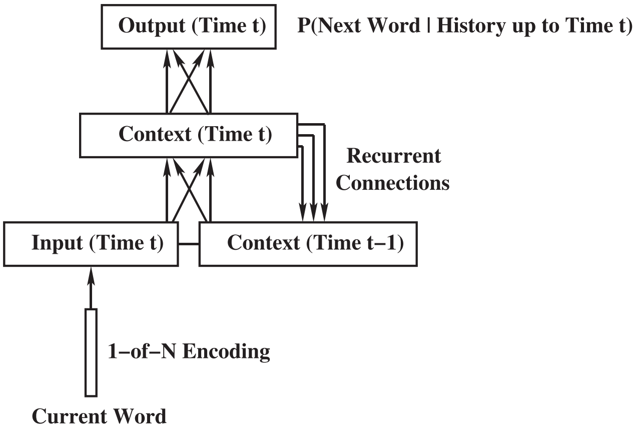


Fig. 9. Recurrent neural network for language modelling.

More recently, it was shown that the basic LSTM node design can be simplified to improve both memory capacity and ease of training (Cho et al., 2014). The alternative reset-update implementation only features two gates, one which allows information to drop when found irrelevant, and one to control how much information is passed along. This design was used to good effect in a statistical machine translation application (Cho et al., 2014). It is thus likely that its advantages would also carry over to more generic language modelling.

Note that there have been other advances in ASR language modeling besides deep neural network language models. For example, we could mention exponential models (such as "model M" (Chen, 2009)), syntactic models (Emami and Jelinek, 2005), Bayesian models (Teh, 2006), etc. For the sake of brevity, we omit further discussion of such techniques because they are largely orthogonal to deep learning.

3.2. *Sparsity promotion*

In recent years, sparsity promoting approaches, perhaps best known from their use in compressive sensing (Candes and Wakin, 2008), have become increasingly popular for pattern recognition and classification tasks. Fundamentally, these techniques leverage knowledge of the test sample to inform the construction of a statistical model on-the-fly. This is usually done by selecting the subset of training instances deemed "most relevant" to the (test) observation at hand.

3.2.1. *Sparse representations*

Sparse representation (SR) methods represent the test sample as a linear combination of pre-specified training instances (sometimes referred to as atoms) collected in a dictionary. The most common applications are in signal compression and reconstruction, where the dictionary obeys a restricted isometric property and is nearly orthornormal. In the speech community, SRs have been used both with instances of individual frames (Gemmeke et al., 2011) and with segments resampled to a fixed-length (Sainath et al., 2011).

The basic approach is to model each D-dimensional (test) observed feature vector $\mathbf{x}$ as:

$$\mathbf{x} = \mathbf{H}\beta, \tag{5}$$

where the $D \times N$ matrix $\mathbf{H}$ is a dictionary of N training instances, and β is a weight vector specifying the best linear combination of these instances that accurately describes $\mathbf{x}$. The dictionary $\mathbf{H}$ is normally overcomplete, in the sense that any observation $\mathbf{x}$ may have several alternative adequate explanations. This ambiguity is resolved by preferring the explanation that involves *the most sparse subset* of dictionary atoms. Ideally, the optimal β is such that only the atoms in $\mathbf{H}$ which belong to the same class as $\mathbf{x}$ have a non-zero weight. The dictionary itself typically comprises a small subset of the complete training set, for example derived via pre-selection with a search based on k nearest neighbours (Sainath et al., 2010). There is a vast literature discussing under which conditions SRs exist and how to efficiently find β: see, e.g. (Elad, 2010) and the references therein.

Once the feature vector is modelled as in (5), classification or recognition proceeds by leveraging labels associated with all relevant training instances. Labels can range from HMM-states on individual frames (Deselaers et al., 2007), to phone classes for phone segments (Sainath et al., 2011), and word labels for word segments (Demuynck et al., 2011). Let us denote by $\mathbf{G}$ the $Q \times N$ binary matrix that associates each instance in $\mathbf{H}$ with one of Q class labels. Then posterior estimates are obtained as:

$$P(q|\mathbf{x}) \propto \mathbf{g}_q\beta, \tag{6}$$

where $\mathbf{g}_q$ represents the row in $\mathbf{G}$ corresponding to class q, effectively picking out those instances associated with that class. For recognition, the posterior estimates (6) can easily be converted to observation likelihoods, as necessary.

3.2.2. *Exemplar-based processing*

The SR methods mentioned above clearly fall within the general category of exemplar-based processing techniques, since they rely on a particular subset of training instances to build a local model specifically for every test sample. A critical limitation, however, is the restriction to fixed-length units, which is imposed by the matrix formalism. A more general approach is to match (variable-length) speech segments to (variable-length) reference templates using quick and approximate *template matching* techniques (De Wachter et al., 2007).

The difference between the two approaches is visualised in Fig. 10. In both cases, processing comprises similar steps of feature extraction (e.g. computing $\mathbf{x}$, or a sequence thereof), exemplar selection (e.g. in SR, specifying $\mathbf{H}$),

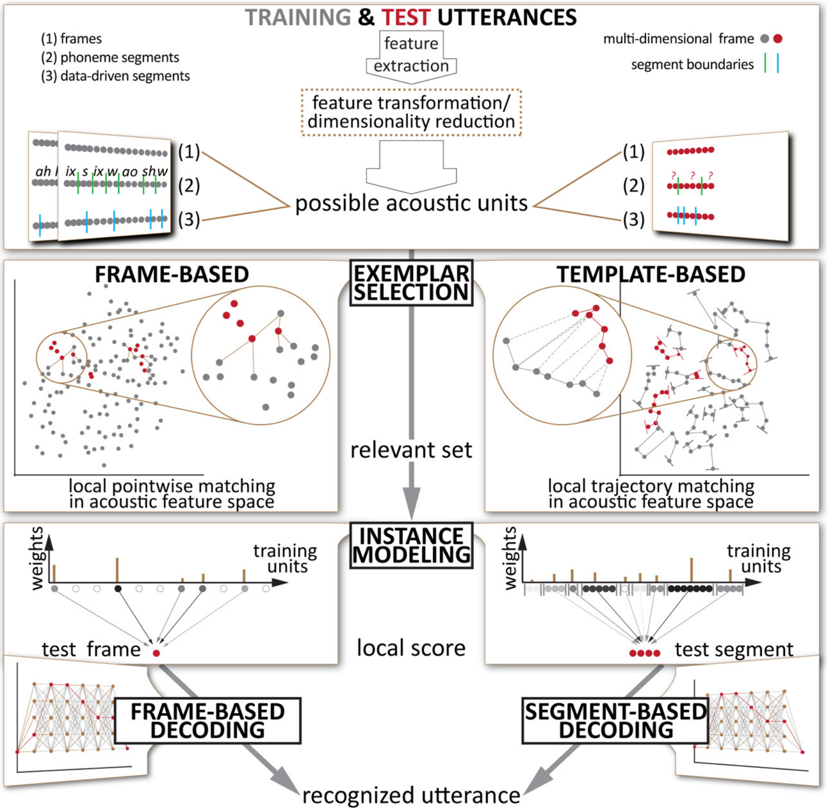


Fig. 10. Frame-level vs. segment-level exemplar-based processing.

instance modelling (e.g. finding the sparse weights β), and decoding (e.g. computing $P(q|\mathbf{x})$ or similar likelihoods). But where SR methods involve local pointwise matching in acoustic feature space, the template-based approach requires local trajectory matching. This is normally done using constrained dynamic time warping (DTW) (Demuynck et al., 2011), which provides the necessary compromise between too little and too much warping, thus explicitly coping with the wide range of temporal variations in speech. Note that, just like classical HMM methods, DTW-based template matching techniques directly calculate the class conditional likelihoods $P(\mathbf{x}|q)$ used for recognition (as opposed to the local posterior probabilities like SR).

As suggested in the upper left of Fig. 10, segments can be either aligned against some existing phoneme inventory or, more generally, derived in a data-driven way. In the latter case, the motivation is to find recurring patterns (units) that help characterise different phone classes. This can be done using latent perceptual mapping (LPM) (Sundaram and Bellegarda, 2010), a technique closely related to the LSM approach mentioned in Section 2.1. In LSM applications like hybrid semantic language modeling (Bellegarda, 1296), a text document is treated as a bag of words. In the same way, in LPM a speech segment is treated as a bag of acoustic units drawn from a limited, data-driven vocabulary. Because that vocabulary can in principle be optimised for the task at hand, this approach was shown to achieve the same level of accuracy as DTW methods but at a *lower dimensionality* (Sundaram and Bellegarda, 2012).

3.3. Knowledge Integration

It has long been recognized that a stark area of deficiency for standard speech front-ends based on MFCC or PLP features lies in fine time structure: typically, energy in the signal is averaged over windows of 25 ms or more, and the time structure at finer scales, which we know is salient to listeners (Licklider, 1951), is lost. This has sparked interest in alternative features, with the potential to achieve complementary descriptions of the spectro-temporal patterns in speech. This in turn has brought to the fore the need to properly integrate into the speech processing workflow a number of possibly heterogeneous sources of information.

3.3.1. *Robust feature extraction*

One avenue of investigation into alternative features is based on the observation that prominent (high-energy) acoustic events tend to exhibit spectro-temporal patterns with very distinctive characteristics. Even when the remaining spectral components of the signals are masked by competing sources, these high-energy patterns stand out in the spectrogram. This has led to the influential "tandem" modelling, which combines a discriminative, trained neural network front-end with mature HMM-based sequence modelling (Ellis et al., 2001). Similarly, deep autoencoder networks, so named because they are trained for auto-associativity, have been successful in providing extra features focused on salient events in so-called bottleneck systems (Grezl et al., 2007).

Another avenue of investigation has recently opened up due to advances in non-invasive *biological data acquisition* systems. Techniques such as Electromagnetic Articulography, offering excellent temporal resolution, and Magnetic Resonance Imaging, offering superior spatial resolution, have enabled a much better analysis of the speech production process (Ramanarayanan et al., 2013). Many insights have thus been gained regarding the spatio-temporal details of speech generation. Using functional Magnetic Resonance Imaging (fMRI), for example, it is now possible to obtain cortical depictions of regional brain activations during execution of language-related tasks. This has underscored, among other things, the critical importance of articulatory features in speech perception (Mesgarani et al., 2014).

Finally, suitable features can also be computed directly from either the linear spectrum or filter bank outputs (and soon perhaps even from the waveform itself, as recently suggested in Tüske et al. (2014)) by adding *convolutional and max-pooling layers* to a traditional DNN. The resulting (deep) convolutional neural network (CNN) has been shown to be competitive with a normal DNN operating on standard MFCC input (Sainath et al., 2013), as long as the configuration is appropriate in terms of convolutional layers, hidden units, and pooling strategy. In particular, the optimal choice of the pooling sizes is determined by the convolution filter design and, more importantly, by the nature of the phonetic space expressed in scaled frequency. Weight sharing can be applied across all time and frequency components, by using a large number of hidden units in the convolutional layers to capture the differences between low and high frequency components (Sainath et al., 2013).

3.3.2. *Heterogeneous information streams*

Given the wide variety of features that can be extracted from the speech signal, attention must be given to the individual processing and recombination of several feature streams, each stream representing a complementary description of the signal. Unfortunately, such streams tend to exhibit varying degrees of asynchrony. It is of course always possible to perform "late" system combination (done at the recognition outcome stage) using a technique such as ROVER (Fiscus, 1997). ROVER aligns the transcription outputs of multiple recognizers in order to form a word lattice; ambiguities in the lattice (alternate explanations for the same period of time) are then resolved by majority voting.

On the other hand, more complex, "early" combination, i.e. done at the modelling stage, may be even more effective. Recently, *segmental conditional random fields* (SCRFs) have emerged as a powerful and fast method to do so (Zweig and Nguyen, 2010). This approach allows for the integration of different feature extraction strategies into a single discriminative framework. When operating on word graphs and with word level features, for example, SCRFs not only allow word-level scores to be combined, but also promote discrete (sub-word) detectors such as the single best phone sequence. This is because detector events can easily be converted to word-level scores, either by means of a Levenshtein distance or by means of automatically detected relations between a word and a phone sequence (Demuynck et al., 2011).

The ultimate goal is to be able to leverage all manners of structural information potentially available, including independent constructed resources such as thesauri and more sophisticated knowledge bases. From a practical point of view, such advances are critical, because the integration of multiple heterogeneous knowledge sources tends to dramatically increase the robustness of the overall system. This in turn allows the deployed system to be successful in a greater variety of situations, be it in terms of external environments, audio channels, speaker accents and/or pathologies, emotional states and communicative styles of the speaker, or discourse domains.

4. Conclusion

In this contribution, we have examined the expanding role of statistical methods in computational language and speech processing, particularly as they pertain to emerging applications like personal assistance and machine translation. The increasingly widespread use of such methods can be traced to the convergence of several factors, including the growing amount of data now accessible, sustained advances in computing power and storage capabilities, and recent improvements in machine learning algorithms. We have reviewed the top three trends that have recently evolved in each of the language and speech communities, along with their most salient characteristics.

4.1. Language processing

In computational language processing, we have discussed the importance of distributional semantics, user-generated content, and evaluation campaigns. All three trends relate to different aspects of statistical modelling in language processing: representation, data, and benchmarking. Computational semantics is one of the most challenging tasks in language processing, and arguably one of most symbolically oriented sub-areas, due to the complexities of semantic representations and inference. Research in distributional semantics explores interesting and promising avenues towards entirely statistical meaning representations. Being able to automatically induce, and potentially adapt, semantic representations can have far-reaching consequences allowing for a much tighter integration of semantics and several applications such as parsing or machine translation than was hitherto possible.

Language processing aims to describe the relevant phenomena occurring in natural language. The emergence of the Internet, and social media in particular, have changed the way we communicate. Analysing user-generated content proves to be a challenging task for many established language processing techniques, which are typically trained on clean and carefully annotated data sets. At the same time, user-generated content offers a wide spectrum of ways in which data can be analysed and exploited. Both aspects together also highlight the need for finding ways to automatically induce representations that are sufficiently general to be generated from data sets with very limited, and even noisy, annotations.

The increasing importance of statistical methods in computational language processing brought also with it the tradition of rigorous evaluation. Many tasks are nowadays connected to recurring evaluations campaigns, which are a driving force in pushing the state of the art in a way such that progress can be quantitatively measured. This means also that evaluation campaigns have to be carefully designed to strike the right balance between what is desirable and what is practical. The recent trend of using crowd-sourcing platforms such as Amazon's Mechanical Turk to annotate data is an interesting development that allows for human annotation in a short amount of time and at a very affordable price. While crowd-sourced annotations still require a significant amount of quality control, filtering, and potentially post-editing, they do offer the possibility for quantitative assessments in a flexible way, which is crucial in order to cover the very wide spectrum of tasks that computational language processing are and will be used for.

4.2. Speech processing

In speech processing, deep learning has recently emerged as a major driver of improvements in modelling accuracy, both for acoustic modelling and language modelling. Sparse representations, and more generally exemplar-based processing, have proven better than traditional models at coping with fine details present in the underlying (unknown) distribution of the data. And new advances in feature extraction, together with a more effective integration of multiple and heterogeneous information streams, have helped make the technology more robust to more speakers in more acoustic environments than ever before.

It is worth noting that these three trends are fundamentally intertwined. Deep learning has pervasive ramifications due to the versatility of the DNN architecture. Because the hierarchy of hidden nodes in a DNN composes a collection of Disjunctive Normal Form (DNF) formulas (one per output node) over the basic patterns detected in the first layer, the network can represent a very large collection of input patterns: every pattern that satisfies one of the DNF formulas at the output nodes. As a result, the network can in effect be viewed as a dictionary, with the benefit that it can represent a much greater number of patterns than any "flat" dictionary that explicitly stores all the structures that can be found in the data, such as those used for SRs (Ranzato et al., 2007). Similarly, it can learn to represent temporal patterns through recurrence structures like LSTM, and convolutive networks (Boureau et al., 2010) can capture spectro-temporal patterns in a data-driven fashion.

On the other hand, sparsity promoting techniques bring to bear the powerful concept of local modelling, specific to each individual test observation. They can thus result in more efficient DNN training, via selection of a subset of "most representative" training cases. In addition, they can potentially help weigh the relative importance of multiple knowledge sources. Conversely, alternative features open the door to new DNN configurations, and the resulting feature spaces may well be more conducive to sparse representations, in turn facilitating sparsity promotion.

4.3. Perspectives

This contribution makes it clear that a wide spectrum of statistical methods is currently in use for both computational language processing and speech processing. While, historically, the two research communities have evolved in a largely independent manner, this common thread opens up a new avenue for tighter collaboration and more cross-fertilisation.

In fact, as statistical methods continue to mature, we can expect a considerable amount of both qualitative user feedback and quantitative real-world data to inform the next generation of algorithms. We therefore see it as inevitable that the processing of textual and acoustic data will become ever more co-dependent, thereby allowing a more systematic exploitation of the multi-faceted latent knowledge available in the context of the various applications considered. The perspective of harnessing these natural synergies between speech and language bodes well for the continued vitality of the field going forward.

References

Abdel-Hamid, O., Jiang, H., 2013, May. Fast speaker adaptation of hybrid NN/HMM model for speech recognition based on discriminative learning of speaker code. In: Proc. Int. Conf. Acoustics, Speech, Signal Processing, Vancouver, Canada, pp. 7942–7946.

Apple Inc., 2011, October. Siri: Your Wish is its Command. http://www.apple.com/ios/siri/

Bahl, L.R., Jelinek, F., Mercer, R.L., 1983, March. A maximum likelihood approach to continuous speech recognition. In: IEEE Trans. Pattern Anal. Mach. Intel. Vol. PAMI-5, No. 2, pp. 179–190.

Bahl, L.R., Brown, P.F., de Souza, P.V., Mercer, R.L., 1986, April. Maximum mutual information estimation of hidden Markov model parameters for speech recognition. Proc. Int. Conf. Acoustics, Speech, Signal Processing, vol. 1., pp. 49–52.

Bellegarda, J.R., 2000 August. Exploiting latent semantic information in statistical language modeling. In: Juang, B.-H., Furui, S. (Eds.), Proceedings of the IEEE, Special Issue on Speech Recognition and Understanding, vol. 88, No. 8., pp. 1279–1296.

Bellegarda, J.R., 2005, September. Latent semantic mapping: a data-driven framework for modeling global relationships implicit in large volumes of data. In: Deng, L., Wang, K., Chou, W. (Eds.), In: Signal Processing Magazine, Special Issue Speech Technol. Systems for Human-Machine Communication, vol. 22, No. 5, pp. 70–80.

Bellegarda, J.R., 2008 March. Latent semantic mapping: principles & applications. In: Juang, B.H. (Ed.), Synthesis Lectures on Speech and Audio Processing #3. Morgan & Claypool Publishers, Fort Collins, CO.

Bengio, Y., Ducharme, R., Vincent, P., 2003. A neural probabilistic language model. J. Mach. Learn. Res. 3, 1137–1155.

Bentivogli, L., Clark, P., Dagan, I., Giampiccolo, D., 2011. The seventh PASCAL recognizing textual entailment challenge. In: Proc. Text Analysis Conference.

Bergsma, S., Van Durme, B., 2013 August. Using conceptual class attributes to characterize social media users. In: Proc. 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Sofia, Bulgaria, pp. 710–720.

Bies, A., Ferguson, M., Katz, K., MacIntyre, R., 1995. Bracketing guidelines for Treebank II style Penn Treebank project. University of Pennsylvania, Philadelphia, PA.

Bikel, D.M., 2004. Intricacies of Collins' parsing model. Comput. Linguist. 30 (4), 479–511.

Bloodgood, M., Callison-Burch, C., 2010, June. Using Mechanical Turk to build machine translation evaluation sets. In: Proc. NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, Los Angeles, CA, pp. 208–211.

Bojar, O., Buck, C., Federmann, C., Haddow, B., Koehn, P., Leveling, J., Monz, C., Pecina, P., Post, M., Saint-Amand, H., Soricut, R., Specia, L., Tamchyna, A., 2014, June. Findings of the 2014 workshop on statistical machine translation. In: Proc. Ninth ACL Workshop on Statistical Machine Translation, Baltimore, MD, pp. 12–58.

Boureau, Y.-L., Bach, F., LeCun, Y., Ponce, J., 2010, June. Learning mid-level features for recognition. In: Proc. IEEE Int. Conf. Comp. Vision Pattern Recognition, San Francisco, CA, pp. 2559–2566.

Bourlard, H., Morgan, N., 1993. Connectionist Speech Recognition: A Hybrid Approach. Kluwer, Norwell, MA.

Brants, T., Popat, A.C., Xu, P., Och, F.J., Dean, J., 2007. Large language models in machine translation. In: Proc. 2007 Joint Conf. Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pp. 858–867.

Brill, E., 1995. Transformation-based error-driven learning and natural language processing: a case study in part-of-speech tagging. Comput. Linguist. 21 (December (4)), 543–565.

Bronner, A., Monz, C., 2012, April. User edits classification using document revision histories. In: Proc. 13th Conf. European Chapter of the Association for Computational Linguistics, Avignon, France, pp. 356–366.

Callison-Burch, C., Dredze, M., 2010. Creating speech and language data with Amazon's Mechanical Turk. In: Proc. of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, Los Angeles, pp. 1–12.

Callison-Burch, C., Koehn, P., Monz, C., Peterson, K., Przybocki, M., Zaidan, O., 2010, July (Revised August 2010. Findings of the 2010 joint workshop on statistical machine translation and metrics for machine translation. In: Proc. Joint Fifth ACL Workshop on Statistical Machine Translation and Metrics MATR, Uppsala, Sweden, pp. 17–53.

Callison-Burch, C., Koehn, P., Monz, C., Zaidan, O., 2011. Findings of the 2011 workshop on statistical machine translation. In: Proc. Sixth ACL Workshop on Statistical Machine Translation, Edinburgh, UK, pp. 22–64.

Candes, E.J., Wakin, M.B., 2008. An introduction to compressive sampling. IEEE Signal Process. Mag. 25 (2), 21–30.

Carter, S., Weerkamp, W., Tsagkias, M., 2013. Microblog language identification: overcoming the limitations of short, unedited and idiomatic text. Lang. Resour. Eval. 47 (1), 195–215.

Cettolo, M., Niehues, J., Stüker, S., Bentivogli, L., Federico, M., 2013. Report on the 10th IWSLT evaluation campaign. In: Proc. 10th International Workshop on Spoken Language Translation.

Chang, K.-W., Yih, W.-T., Meek, C.C., 2013, October. Multi-relational latent semantic analysis. In: Proc. 2013 ACL Conf. Empirical Methods in Natural Language Processing, Seattle, WA.

Chen, S.F., Goodman, J., 1999. An empirical study of smoothing techniques for language modeling. Comput. Speech Lang. 13 (4), 359–393.

Chen, S.F., 2009, June. Shrinking exponential language models. In: Proc. 2009 Conf. Human Language Technologies: North American Chapter of the Association for Computational Linguistics, Boulder, CO.

Chierchia, G., 1992. Anaphora and dynamic binding. Linguist. Philos. 15, 111–183.

Cho, K., van Merriënboer, B., Gulcehre, Bahdanau, D., Bougares, C., Schwenk, F., Bengio, H.Y., 2014 July. Learning phrase representations using RNN encoder decoder for statistical machine translation. In: Proc. 2014 ACL Conf. Empirical Methods in Natural Language Processing, Doha, Qatar, pp. 1724–1734.

Collins, M., 1999. Head-Driven Statistical Models for Natural Language Parsing (PhD thesis). University of Pennsylvania, Philadelphia, PA.

Daxenberger, J., Gurevych, I., 2013, October. Automatically classifying edit categories in Wikipedia revisions. In: Proc. 2013 ACL Conf. Empirical Methods in Natural Language Processing, Seattle, WA, pp. 578–589.

De Wachter, M., Matton, M., Demuynck, K., Wambacq, P., Cools, R., Van Compernolle, D., 2007. Template based continuous speech recognition. IEEE Trans. Audio Speech Lang. Process. 15 (May), 1377–1390.

Dean, J., Corrado, G.S., Monga, R., Chen, K., Devin, M., Le, Q.V., Mao, M.Z., Ranzato, M., Senior, A., Tucker, P., Yang, K., Ng, A.Y., 2012 December. Large scale distributed deep networks. In: Proc. Advances in Neural Information Processing Systems (NIPS 2012), Lake Tahoe, CA.

Dehak, N., Kenny, P., Dehak, R., Dumouchel, P., Ouellet, P., 2011. Front-end factor analysis for speaker verification. IEEE Trans. Audio Speech Lang. Process. 19 (May (4)), 788–798.

Dekker, P., 2000. Coreference and representationalism. In: von Heusinger, K., Egli, U. (Eds.), Reference and Anaphorical Relations. Kluwer, Dordrecht, The Netherlands, pp. 287–310.

Demuynck, K., Seppi, D., Van Compernolle, D., Nguyen, P., Zweig, G., 2011, May. Integrating meta-information into exemplar-based speech recognition with segmental conditional random fields. In: Proc. Int. Conf. Acoustics, Speech, Signal Processing, Prague, Czech Republic.

Demuynck, K., Seppi, D., Van Hamme, H., Van Compernolle, D., 2011 May. Progress in example-based automatic speech recognition. In: Proc. Int. Conf. Acoustics, Speech, Signal Processing, Prague, Czech Republic.

Deselaers, T., Heigold, G., Ney, H., 2007. Speech recognition with state-based nearest neighbour classifiers. In: Proc. 8th Ann. Conf. Int. Speech Comm. Assoc. (InterSpeech), Antwerp, Belgium.

Eidelman, V., Hollingshead, K., Resnik, P., 2011 July. Noisy SMS machine translation in low-density languages. In: Proc. Sixth ACL Workshop on Statistical Machine Translation, Edinburgh, UK, pp. 344–350.

Eisenstein, J., 2013, June. What to do about bad language on the Internet. In: Proc. 2013 Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Atlanta, GA, pp. 359–369.

Elad, M., 2010. Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing. Springer-Verlag.

Ellis, D.P.W., Singh, R., Sivadas, S., 2001. Tandem acoustic modeling in large-vocabulary recognition. In: Proc. IEEE Int. Conf. Acous. Speech, Sig. Proc., Salt Lake City, UT I-517-520.

Emami, A., Jelinek, F., 2005. A neural syntactic language model. Mach. Learn. 60, 195–227.

Fellbaum, C. (Ed.), 1998. WordNet: An Electronical Lexical Database. MIT Press.

Finin, T., Murnane, W., Karandikar, A., Keller, N., Martineau, J., Dredze, M., 2010 June. Annotating named entities in twitter data with crowd sourcing. In: Proc. of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, Los Angeles, pp. 80–88.

Fiscus, J., 1997 December. A post-processing system to yield reduced word error rates: recognizer output voting error reduction (ROVER). In: Proc. IEEE Aut. Speech Recog. Understand. Workshop (ASRU), Santa Barbara, CA.

Furui, S., 1981. Cepstral analysis technique for automatic speaker verification. IEEE Trans. Acoust. Speech Signal Process. 29, 254–272.

Gao, J., Nie, J.-Y., Zhou, M., 2006. Statistical query translation models for cross-language information retrieval. ACM Trans. Asian Lang. Inf. Process. 5 (December (4)), 323–359.

Gemmeke, J.F., Cranen, B., Remes, U., 2011. Sparse imputation for large vocabulary noise robust ASR. Comput. Speech Lang. 25 (2), 462–479.

Google Inc., 2012. Google Now: Just the right information at just the right time. http://www.google.com/landing/now/

Gormley, M., Gerber, A., Harper, M., Dredze, M., 2010 June. Non-expert correction of automatically generated relation annotations. In: Proc. of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, Los Angeles, pp. 204–207.

Grezl, F., Karafiat, M., Kontar, S., Cernocky, J., 2007 March. Probabilistic and bottleneck features for LVCSR of meetings. In: Proc. Int. Conf. Acoustics Speech, Signal Processing, Honolulu, HI.

Guo, W., Li, H., Ji, H., Diab, M., 2013 August. Linking tweets to news: a framework to enrich short text data in social media. In: Proc. 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Sofia, Bulgaria, pp. 239–249.

Gupta, V., Kenny, P., Ouellet, P., Stafylakis, T., 2014 May. I-vector-based speaker adaptation of deep neural networks for French broadcast audio transcription. In: Proc. Int. Conf. Acoustics Speech, Signal Processing, Florence, Italy.

Hassan, H., Menezes, A., 2013 August. Social text normalization using contextual graph random walks. In: Proc. 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Sofia, Bulgaria, pp. 1577–1586.

Heigold, G., McDermott, E., Vanhoucke, V., Senior, A., Bacchiani, M., 2014 May. Asynchronous stochastic optimization for sequence training of deep neural networks. In: Proc. Int. Conf. Acoustics, Speech, Signal Processing, Florence, Italy.

Heilman, M., Smith, N., 2010 June. Rating computer-generated questions with Mechanical Turk. In: Proc. of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, Los Angeles, pp. 35–40.

Hermansky, H., 1990. Perceptual linear predictive (PLP) analysis of speech. J. Acoust. Soc. Am. 87 (4), 1738–1752.

Hinton, G.E., Osindero, S., Teh, Y., 2006. A fast learning algorithm for deep belief nets. Neural Comput. 18, 1527–1554.

Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T.N., Kingsbury, B., 2012. Deep neural networks for acoustic modeling in speech recognition. IEEE Signal Process. Mag. 29 (November (6)), 82–97.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.

Hong, J., Baker, C., 2011. How good is the crowd at "real" WSD? In: Proc. of the 5th Linguistic Annotation Workshop, pp. 30–37.

Hovy, E., Lin, C.Y., 1997. Automated text summarization in SUMMARIST. In: Proc. Intelligent Scalable Text Summarization Workshop, pp. 18–24.

Jehl, L., Hieber, F., Riezler, S., 2012 June. Twitter translation using translation-based cross-lingual retrieval. In: Proc. Seventh ACL Workshop on Statistical Machine Translation, Montréal, Canada, pp. 410–421.

Joachims, T., 2002. Optimizing search engines using click through data. In: Proc. Eighth ACM SIGKDD International Conf. Knowledge Discovery and Data Mining KDD'02, New York, NY, pp. 133–142.

Kamp, H., Reyle, U., 1993. From Discourse to Logic. Kluwer, Dordrecht, The Netherlands.

Kastner, I., Monz, C., 2009 March. Automatic single-document key fact extraction from newswire articles. In: Proc. 12th Conf. European Chapter of the ACL (EACL 2009), Athens, Greece, pp. 415–423.

Kim, Y., 2014 October. Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, pp. 1746–1751.

Kingsbury, B., Sainath, T.N., Soltau, H., 2012 September. Scalable minimum Bayes risk training of deep neural network acoustic models using distributed Hessian-free optimization. In: Proc. 13th Ann. Conf. Int. Speech Comm. Assoc. (InterSpeech), Portland, OR.

Knight, K., Marcu, D., 2001. Summarization beyond sentence extraction: a probabilistic approach to sentence compression. Artif. Intell. 13 (1), 91–107.

Koehn, P., 2009. Statistical Machine Translation. Cambridge University Press.

Lampos, V., Preoiuc-Pietro, D., Cohn, T., 2013 August. A user-centric model of voting intention from social media. In: Proc. 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Sofia, Bulgaria, pp. 993–1003.

Lapata, M., Keller, F., 2005. Web-based models for natural language processing. ACM Trans. Speech Lang. Process. 2, 1–31.

Larochelle, H., Erhan, D., Courville, A., Bergstra, J., Bengio, Y., 2007. An empirical evaluation of deep architectures on problems with many factors of variation. In: Proc. 24th Int. Conf. Machine Learning, pp. 473–480.

Lawson, N., Eustice, K., Perkowitz, M., Yetisgen-Yildiz, M., 2010 June. Annotating large email datasets for named entity recognition with Mechanical Turk. In: Proc. of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, Los Angeles, pp. 71–79.

Levy, R., Manning, C.D., 2003 July. Is it harder to parse Chinese, or the Chinese Treebank? In: Proc. 41st Annual Meeting of the Association for Computational Linguistics, Sapporo, Japan, pp. 439–446.

Licklider, J., 1951. A duplex theory of pitch perception. Cell. Mol. Life Sci. 7, 128–134.

Ling, W., Xiang, G., Dyer, C., Black, A., Trancoso, I., 2013 August. Microblogs as parallel corpora. In: Proc. 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Sofia, Bulgaria, pp. 176–186.

Ling, W., Dyer, C., Black, A.W., Trancoso, I., 2013 October. Paraphrasing 4 microblog normalization. In: Proc. 2013 ACL Conf. Empirical Methods in Natural Language Processing, Seattle, WA, pp. 73–84.

Liu, X., Duh, K., Matsumoto, Y., Iwakura, T., 2014. Learning character representations for Chinese word segmentation. In: NIPS Workshop on Modern Machine Learning and Natural Language Processing.

Lopez, A., 2008. Statistical machine translation. ACM Comput. Surv. 40 (August (3)), 8:1–8:49.

Marcus, M.P., Santorini, B., Marcinkiewicz, M.A., 1993. Building a large annotated corpus of English: The Penn Treebank. Comput. Linguist. 19 (2), 313–330.

Marcus, M., 1994. New trends in natural language processing: statistical natural language processing. In: Roe, D.B., Wilpon, J.G. (Eds.), Voice Communication Between Humans and Machines. National Academy Press, Washington, DC, pp. 482–504.

Marge, M., Banerjee, S., Rudnicky, A., 2010 June. Using the Amazon Mechanical Turk to transcribe and annotate meeting speech for extractive summarization. In: Proc. of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, Los Angeles, pp. 99–107.

Mesgarani, N., Cheung, C., Johnson, K., Chang, E., 2014 January. Phonetic feature encoding in human superior temporal gyrus. Science, No. 1245994.

Microsoft Corp., 2014. Windows Phone, Meet Cortana, your very own personal assistant. http://www.windowsphone.com/en-us

Mikolov, T., Karafiat, M., Burget, L., Cernocky, J., Khudanpur, S., 2010 September. Recurrent neural network based language model. In: Proc. 11th Ann. Conf. Int. Speech Comm. Assoc. (InterSpeech), Makuhari, Japan.

Mikolov, T., Yih, W.-T., Zweig, G., 2013 June. Linguistic regularities in continuous space word representations. In: Proc. 2013 Conf. North American Chapter Association for Computational Linguistics, Atlanta, GA.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J., 2013 December. Distributed representations of words and phrases and their compositionality. In: Proc. Advances in Neural Information Processing Systems 26 (NIPS 2013), Lake Tahoe, CA.

Mitchell, J., Lapata, M., 2008 June. Vector-based models of semantic composition. In: Proc. Conf. Association for Computational Linguistics, Columbus, OH, pp. 236–244.

Mitchell, J., Lapata, M., 2010. Composition in distributional models of semantics. Cognit. Sci. 34 (8), 1388–1439.

Montague, R., 1970. Universal grammar. Theoria 36, 373–398.

Muskens, R., 1996. Combining Montague semantics and discourse representation. Linguist. Philos. 19 (2), 143–186.

Nastase, V., Strube, M., 2013. Transforming wikipedia into a large scale multilingual concept network. Artif. Intell. 194, 62–85.

Negri, M., Mehdad, Y., 2010 June. Creating a bi-lingual entailment corpus through translations with Mechanical Turk: $100 for a 10-day rush. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, Los Angeles, pp. 212–216.

Negri, M., Bentivogli, L., Mehdad, Y., Giampiccolo, D., Marchetti, A., 2011 July. Divide and conquer: crowd sourcing the creation of cross-lingual textual entailment corpora. In: Proc. 2011 ACL Conf. Empirical Methods in Natural Language Processing, Edinburgh, UK, pp. 670–679.

Ng, H.T., Wu, S.M., Wu, Y., Hadiwinoto, C., Tetreault, J., 2013 August. The conll-2013 shared task on grammatical error correction. In: Proc. Seventeenth Conf. Computational Natural Language Learning: Shared Task, Sofia, Bulgaria, pp. 1–12.

Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., Smith, N.A., 2013 June. Improved part-of-speech tagging for online conversational text with word clusters. In: Proc. 2013 Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Atlanta, GA, pp. 380–390.

Partee, B., 1984. Compositionality. In: Landman, F., Veltman, F. (Eds.), Varieties of Formal Semantics. Kluwer, Dordrecht, The Netherlands, pp. 281–312.

Pennington, J., Socher, R., Manning, C., 2014 October. Glove: global vectors for word representation. In: Proc. 2014 ACL Conf. Empirical Methods in Natural Language Processing, Doha, Qatar, pp. 1532–1543.

Ponzetto, S.P., Strube, M., 2007. Knowledge derived from wikipedia for computing semantic relatedness. J. Artif. Intell. Res. 30 (October (1)), 181–212.

Povey, D., Kanevsky, D., Kingsbury, B., Ramabhadran, B., Saon, G., Visweswariah, K., 2008 April. Boosted MMI for model and feature-space discriminative training. In: Proc. Int. Conf. Acoustics, Speech, Signal Processing, Las Vegas, NV, pp. 4057–4060.

Povey, D., 2003. Discriminative Training for Large Vocabulary Speech Recognition (PhD Dissertation). University of Cambridge, Cambridge, UK.

Rabiner, L.R., Juang, B.H., Lee, C.-H., 1996. An overview of automatic speech recognition. In: Lee, C.-H., Soong, F.K., Paliwal, K.K. (Eds.), Automatic Speech and Speaker Recognition: Advanced Topics. Kluwer Academic Publishers, Boston, MA, pp. 1–30 (Chapter 1).

Ramanarayanan, V., Goldstein, L., Narayanan, S., 2013. Spatio-temporal articulatory movement primitives during speech production: extraction, interpretation, and validation. J. Acoust. Soc. Am. 134 (2), 1378–1394.

Ranzato, M.A., Boureau, Y.-L., LeCun, Y., 2007. Sparse feature learning for deep belief networks. In: Proc. Neural Information Processing Systems.

Resnik, P., Smith, N.A., 2003. The web as a parallel corpus. Comput. Linguist. 29 (September (3)), 349–380.

Riezler, S., Liu, Y., 2010. Query rewriting using monolingual statistical machine translation. Comput. Linguist. 36 (3), 569–582.

Sainath, T.N., Ramabhadran, B., Nahamoo, D., Kanevsky, D., Sethy, A., 2010. Exemplar-based sparse representation features for speech recognition. In: Proc. 11th Ann. Conf. Int. Speech Comm. Assoc. (InterSpeech), Makuhari, Japan.

Sainath, T.N., Nahamoo, D., Kanevsky, D., Ramabhadran, B., Shah, P.M., 2011 December. A convex hull approach to sparse representations for exemplar-based speech recognition. In: in Proc. IEEE Aut. Specch Recog. Understand. Workshop (ASRU), Waikoloa, HI.

Sainath, T.N., Ramabhadran, B., Nahamoo, D., Kanevsky, D., Van Compernolle, D., Demuynck, K., Gemmeke, J.F., Bellegarda, J.R., Sundaram, S., 2012 November. Exemplar-based processing for speech recognition. IEEE Signal Proces. Mag. 29 (6), 98–113.

Sainath, T.N., Mohamed, A., Kingsbury, B., Ramabhadran, B., 2013 May. Deep convolutional neural networks for LVCSR. In: Proc. Int. Conf. Acoustics, Speech, Signal Processing, Vancouver, Canada.

Saon, G., Soltau, H., Nahamoo, D., Picheny, M., 2013 December. Speaker adaptation of neural network acoustic models using i-vectors. In: Proc. IEEE Aut. Specch Recog. Understand. Workshop (ASRU), Olomouc, Czech Republic, pp. 55–59.

Schmid, H., 1994. Probabilistic part-of-speech tagging using decision trees. In: International Conf. New Methods in Language Processing, Manchester, UK, pp. 44–49.

Schwenk, H., Gauvain, J.-L., 2002. Connectionist language modeling for large vocabulary continuous speech recognition. In: Proc. Int. Conf. Acoustics, Speech, Signal Processing, Orlando, FL, vol. 1, I-765-I-768.

Schwenk, H., 2007. Continuous space language models. Comput. Speech Lang. 21, 492–518.

Seide, F., Li, G., Chen, X., Yu, D., 2011 December. Feature engineering in context-dependent deep neural networks for conversational speech transcription. In: in Proc. IEEE Aut. Speech Recog. Understand. Workshop (ASRU), Waikoloa, HI.

Seide, F., Fu, H., Droppo, J., Li, G., Yu, D., 2014 May. On parallelizability of stochastic gradient descent for speech DNNs. In: Proc. Int. Conf. Acoustics, Speech, Signal Processing, Florence, Italy.

Seide, F., Fu, H., Droppo, J., Li, G., Yu, D., 2014 September. 1-bit stochastic gradient descent and application to data-parallel distributed training of speech DNNs. In: Proc. 15th Ann. Conf. Int. Speech Comm. Assoc. (InterSpeech), Singapore.

Sinha, S., Dyer, C., Gimpel, K., Smith, N.A., 2013. Predicting the NFL using Twitter. In: Proc. ECML/PKDD Workshop on Machine Learning and Data Mining for Sports Analytics, pp. 1–11.

Siniscalchi, S.M., Li, J., Lee, C.-H., 2012 September. Hermitian-based hidden activation functions for adaptation of hybrid HMM/ANN models. In: in Proc. 13th Ann. Conf. Int. Speech Comm. Assoc. (InterSpeech), Portland, OR, pp. 526–529.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C.D., Ng, A., Potts, C., 2013 October. Recursive deep models for semantic compositionality over a sentiment treebank. In: Proc. 2013 ACL Conf. Empirical Methods in Natural Language Processing, Seattle, WA, pp. 1631–1642.

Stymne, S., 2011 July. Spell checking techniques for replacement of unknown words and data cleaning for Haitian Creole SMS translation. In: Proc. Sixth ACL Workshop on Statistical Machine Translation, Edinburgh, UK, pp. 470–477.

Sundaram, S., Bellegarda, J.R., 2010 September. Latent perceptual mapping: a new acoustic modeling framework for speech recognition. In: Proc. 11th Ann. Conf. Int. Speech Comm. Assoc. (InterSpeech), Makuhari, Japan, pp. 881–884.

Sundaram, S., Bellegarda, J.R., 2012 March. Latent perceptual mapping with data-driven variable-length acoustic units for template-based speech recognition. In: Proc. Int. Conf. Acoustics, Speech, Signal Processing, Kyoto, Japan.

Sundermeyer, M., Schluter, R., Ney, H., 2012 September. LSTM neural networks for language modeling. In: Proc. 13th Ann. Conf. Int. Speech Comm. Assoc. (InterSpeech), Portland, OR.

Surdeanu, M., 2013. Overview of the tac2013 knowledge base population evaluation: English slot filling and temporal slot filling. In: Proc. Text Analysis Conference.

Tüske, Z., Golik, P., Schlüter, R., Ney, H., 2014 September. Acoustic modeling with deep neural networks using raw time signal for LVCSR. In: Proc. 15th Ann. Conf. Int. Speech Comm. Assoc. (InterSpeech), Singapore.

Teh, Y.W., 2006 July. A hierarchical Bayesian LM based on Pitman-Yor Processes. In: Proc. Association for Computational Linguistics, Sydney, Australia.

Turney, P.D., Pantel, P., 2010. From frequency to meaning: vector space models of semantics. J. Artif. Intel. Res. 37 (January (1)), 141–188.

Veselý, K., Ghoshal, A., Burget, L., Povey, D., 2013 August. Sequence-discriminative training of deep neural networks. In: Proc. 14th Ann. Conf. Int. Speech Comm. Assoc. (InterSpeech), Lyon, France.

Williams, J.D., Melamed, I.D., Alonso, T., Hollister, B., Wilpon, J., 2011 December. Crowd-sourcing for difficult transcription of speech. In: Proc. IEEE Aut. Speech Recog. Understand. Workshop (ASRU), Waikoloa, HI.

Wu, Y., Yamamoto, H., Lu, X., Matsuda, S., Hori, C., Kashioka, H., 2012 December. Factored recurrent neural network language model in TED lecture transcription. In: Proc. Int. Workshop Spoken Language Translation, Hong Kong, China.

Yang, Y., Eisenstein, J., 2013 October. A log-linear model for unsupervised text normalization. In: Proc. 2013 ACL Conf. Empirical Methods in Natural Language Processing, Seattle, WA, pp. 61–72.

Yao, K., Yu, D., Seide, F., Su, H., Deng, L., Gong, Y., 2012 December. Adaptation of context-dependent deep neural networks for automatic speech recognition. In: Proc. IEEE Spoken Language Technology Workshop (SLT), Berkeley, CA, pp. 366–369.

Yih, W.-T., Zweig, G., Platt, J., 2012 July. Polarity inducing latent semantic analysis. In: Proc. 2012 ACL Conf. Empirical Methods in Natural Language Processing, Jeju Island, Korea.

Yu, D., Yao, K., Su, H., Li, G., Seide, F., 2013 May. KL-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition. In: Proc. Int. Conf. Acoustics, Speech, Signal Processing, Vancouver, Canada.

Yu, D., Deng, L., Seide, F., 2013. The deep tensor neural network with applications to large vocabulary speech recognition. IEEE Trans. Audio Speech Lang. Process. 21 (February (2)), 388–396.

Zajic, D., Dorr, B., Schwartz, R., Monz, C., Lin, J., 2005. A sentence-trimming approach to multi-document summarization. In: Proc. EMNLP 2005 Workshop on Text Summarization.

Zhu, Q., Chen, B., Morgan, N., Stolcke, A., 2005. Tandem Connectionist Feature Extraction for Conversational Speech Recognition. In: Mach. Learning Multimodal Interaction, Lecture Notes Comp. Science., pp. 223–231.

Zweig, G., Nguyen, P., 2010 September. SCARF: a segmental conditional random field toolkit for speech recognition. In: Proc. 11th Ann. Conf. Int. Speech Comm. Assoc. (InterSpeech), Makuhari, Japan.