Natural Language Processing 1

Recent advances and summary of the course

Katia Shutova

ILLC
University of Amsterdam

Outline.

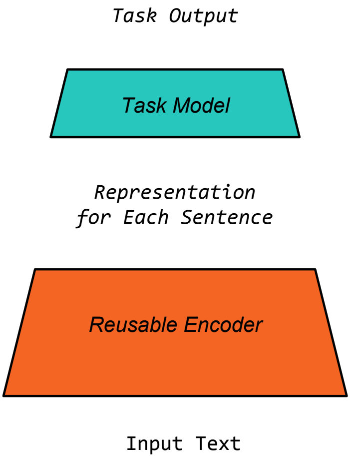
Recent advances in NLP

Summary of the course

Large language models

Paradigm shift:

- instead of training task-specific models
- train a **general-purpose** neural network **sentence encoder**
- which can be applied across diverse NLP tasks.



Why is this useful?

1. Improve performance
   - produce **rich semantic representations** for downstream NLP tasks
2. Improve data efficiency
   - provide a model of sentence representation for language understanding tasks which **lack training data**

What can we expect this model to capture?

- Lexical semantics and meaning disambiguation in context
- Word order
- Some syntactic structure
- Semantic composition
- Idiomatic/non-compositional phrase meanings
- Connotation and social meaning.

ELMo: Embeddings from Language Models

Peters et al. 2018. *Deep contextualized word representations*

- Pretrain a biLSTM model in the language modelling task
- Model context in both directions, produce contextualised word representations
- Use them as input to a task-specific model.

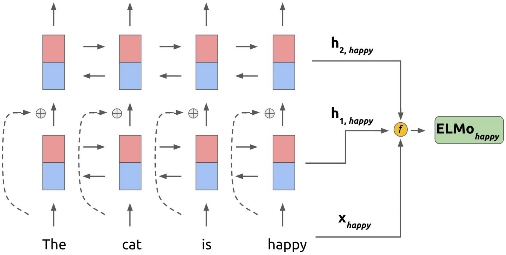


Image credit: Victor Zuanazzi

The ELMo model

Pretraining:

- The encoder is a 2 layer BiLSTM
- The model is trained with the language modelling objective
- jointly maximize log likelihood of the forward and backward directions.

Application:

- ELMo word representations: weighted sum of hidden representations at all layers
- Weights are learned in a given task.

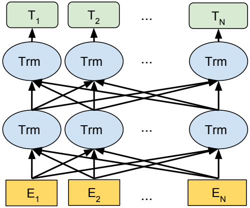
The contributions of ELMo

- **Contextualised word representations** provide a level of disambiguation
- **Deep** representations allow to capture linguistic information at various levels (syntax – lower layers; semantics – higher layers)
- (Large) performance improvements in many NLP tasks
- **Paradigm shift** towards sentence encoder pretraining
- Started the rich history of naming LMs based on Sesame Street characters.

The rise of the Transformer

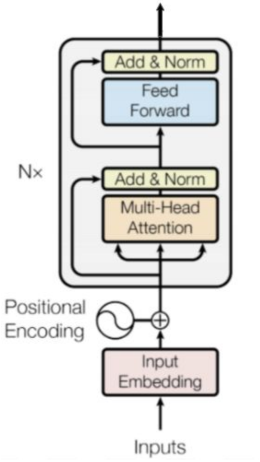
Devlin et al. 2019. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*

- Transformer architecture
- Bidirectional context representation
- Two pretraining tasks: masked language modelling (MLM) and next sentence prediction (NSP)
- Pretrain the encoder and then fine-tune it for a specific task.



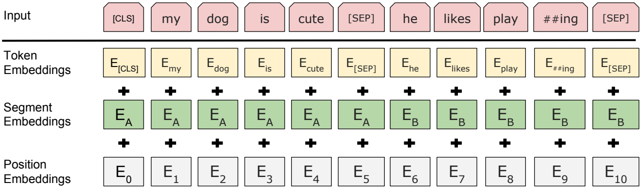
BERT: Architecture

- Stacked Transformer blocks (multi-head attention followed by feed-forward neural network)
- BASE model: 12 Transformer layers, 8 attention heads (110M params)
- LARGE model: 24 Transformer layers, 12 attention heads (340M parameters)



BERT: Input representations

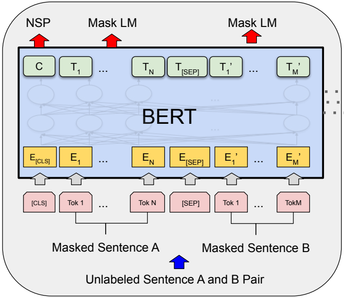
- Introduce special [CLS] and [SEP] tokens
- The [CLS] token represents the whole input sequence
- The [SEP] token indicates a boundary between two segments
- **Input representations** are a sum of token embeddings + position embeddings + segment embeddings.



BERT: Pretraining tasks

Masked language modelling

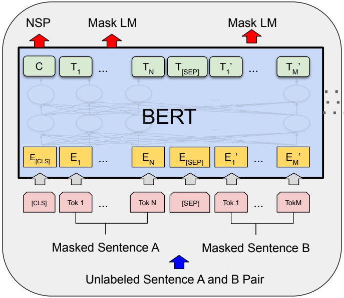
- standard conditional language models only model context in one direction at a time
- BERT performs bidirectional encoding by masking 15% of the input tokens
- Inspired by the cloze task



BERT: Pretraining tasks

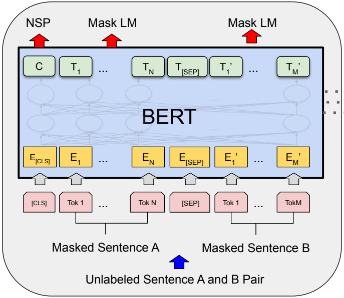
Next sentence prediction

- Randomly sample sentence pairs, such that 50% of the time the sentences follow each other.
- Predict whether the second sentence follows the first or not.
- This models the relations between sentences (useful for many tasks, e.g. QA)

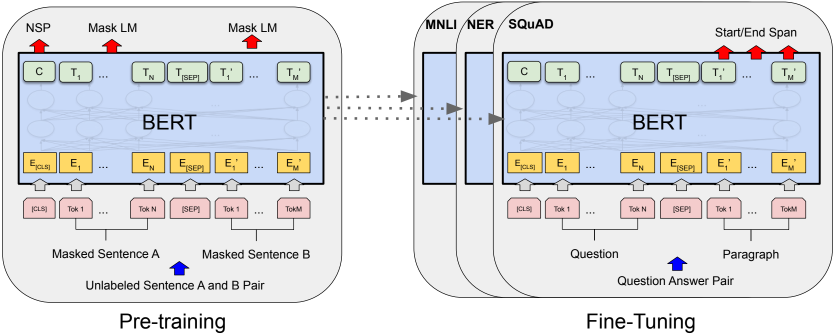


BERT: pretraining

- **Pre-training loss**: the sum of the mean MLM likelihood and the mean NSP likelihood
- **Data**: BooksCorpus (800M words) and English Wikipedia (2500M words)



BERT: fine-tuning



The contributions of BERT

- **Advanced the state-of-the-art** in a range of NLP tasks
- Demonstrated the importance of bidirectional pretraining
- Reduced the need for task-specific architectures
- Most widely-used NLP model (54K+ citations)
- Traditional linguistic hierarchy emerges within layers of BERT (Tenney et al. 2019)
- lower layers – syntax; higher layers – semantics and discourse.

Tenney et al. 2019. *BERT Rediscovers the Classical NLP Pipeline*

Generative language models: The GPT family

Radford et al. 2019. *Language Models are Unsupervised Multitask Learners*

GPT, GPT2, GPT3

- Left-to-right language model
- Generative model, i.e. able to generate text (unlike BERT)
- Transformer architecture (GPT comparable in size to BERT BASE)
- Interesting intuition: multitask learning from natural language instructions.

More than a language model?

- Many tasks are already described in the data in some way
- Can language models learn to perform tasks from natural language instructions found in web text?

> If listened carefully at 29:55, a conversation can be heard between two guys in French: "**-Comment on fait pour aller de l'autre coté? -Quel autre coté?**", which means "**- How do you get to the other side? - What side?**".
>
> If this sounds like a bit of a stretch, consider this question in French: **As-tu aller au cinéma?**, or **Did you go to the movies?**, which literally translates as Have-you to go to movies/theater?
>
> "**Brevet Sans Garantie Du Gouvernement**", translated to English: "**Patented without government warranty**".

InstructGPT and ChatGPT

InstructGPT

- trained to follow an instruction in a prompt and provide a detailed response.

ChatGPT

- optimized for dialogue
- make GPT generations more "conversational": can provide more natural answers, answer follow-up questions etc.

Outstanding challenges and future directions

- Interpretability
- Multitask-learning
- Continual learning
- Low-resource languages
- Few-shot learning and generalisation
- Common sense reasoning

We discuss these topics in an advanced NLP courses, such as *Advanced Topics on Computational Semantics* (block 5)

Outline.

Recent advances in NLP

Summary of the course

Levels of language analysis

1. Morphology — the structure of words.
2. Syntax — the way words are used to form phrases.
3. Semantics
   - Lexical semantics — the meaning of individual words.
   - Compositional semantics — the construction of meaning of longer phrases and sentences (based on syntax).
4. Discourse and pragmatics — meaning in context.

Ambiguity

Ambiguity: *same strings can mean different things*

- Morphology: unionised (*un- ion -ise -ed* vs. *union -ise -ed*)
- Word senses: bank (finance or river?)
- Part of speech: chair (noun or verb?)
- Syntactic structure: I saw a man with a telescope
- Discourse relations: Max fell. John pushed him.

Ambiguity

Ambiguity: *same strings can mean different things*

- Morphology: unionised (*un- ion -ise -ed* vs. *union -ise -ed*)
- Word senses: bank (finance or river?)
- Part of speech: chair (noun or verb?)
- Syntactic structure: I saw a man with a telescope
- Discourse relations: Max fell. John pushed him.

Ambiguity

Ambiguity: *same strings can mean different things*

- Morphology: unionised (*un- ion -ise -ed* vs. *union -ise -ed*)
- Word senses: bank (finance or river?)
- Part of speech: chair (noun or verb?)
- Syntactic structure: I saw a man with a telescope
- Discourse relations: Max fell. John pushed him.

Ambiguity

Ambiguity: *same strings can mean different things*

- Morphology: unionised (*un- ion -ise -ed* vs. *union -ise -ed*)
- Word senses: bank (finance or river?)
- Part of speech: chair (noun or verb?)
- Syntactic structure: I saw a man with a telescope
- Discourse relations: Max fell. John pushed him.

Ambiguity

Ambiguity: *same strings can mean different things*

- Morphology: unionised (*un- ion -ise -ed* vs. *union -ise -ed*)
- Word senses: bank (finance or river?)
- Part of speech: chair (noun or verb?)
- Syntactic structure: I saw a man with a telescope
- Discourse relations: Max fell. John pushed him.

Modelling morphology

unionised: *un- ion -ise -ed* vs. *union -ise -ed*

- stemming, i.e. removing inflections

  unionise

- lemmatisation, i.e. full morphological analysis

  unionise PAST VERB

Modelling morphology

How?

1. Traditionally, rule-based methods
2. More recently, neural models: e.g. character LSTMs (advanced NLP courses)

Why is it useful?

- provides information about word structure, e.g. *shame -less*. Relevant to semantics.
- and grammatical properties, e.g. part of speech, tense, number. Informative for syntactic tasks.

Modelling syntax

How?

1. **n-gram** language models
   - compute probability of a sequence
2. **Part-of-speech** tagging
   - Sequence labelling task (assign a label to each word)
   - Hidden Markov Models (HMM)
   - more recently, neural sequence labelling (e.g. LSTMs)
3. Syntactic **parsing**
   - (Probabilistic) context-free grammars
   - Chart parsing
   - Dependency structure

Modelling syntax

What kind of information do they capture?

1. **n-gram** language models
   - word order
   - short-distance dependencies
2. **Part-of-speech** tagging
   - grammatical properties of words
   - coarse-grained word sense
3. Syntactic **parsing**
   - hierarchical structure of sentences
   - dependencies between words
   - types of phrases (e.g. NP, VP).

Modelling syntax

Why is this useful?

1. **n-gram** language models
   - language generation, e.g. fluency ranking
   - speech recognition, i.e. hypothesis ranking
   - as features in classification tasks
2. **Part-of-speech** tagging
   - precursor to parsing
   - lexical semantics
   - as features in classification tasks
3. Syntactic **parsing**
   - semantic composition
   - co-reference resolution (to identify NPs)
   - applications (e.g. summarisation).

Modelling semantics

How?

1. **Lexical** semantics
   - word sense disambiguation (supervised classification)
   - distributional semantics
   - skip-gram word embeddings
2. **Compositional** semantics
   - compositional distributional semantics
   - neural models: LSTMs and tree LSTMs

Which of the above models rely on syntax?

Modelling semantics

What kind of information do these models capture?

1. **Lexical** semantics
   - word meanings / senses
   - semantic similarity
   - semantic relations (e.g. hyponymy, synonymy)
2. **Compositional** semantics
   - meanings of phrases
   - sentence representation learning
     (general-purpose representations useful for many tasks – underlie SOTA models; discussed in ATCS course)

Modelling semantics

Why is this useful?

1. **Lexical** semantics
   - in applications (e.g. sentiment, summarisation)
   - in parsing (e.g. to resolve PP attachment ambiguity)
   - semantic similarity useful in co-reference resolution
   - input to neural models
2. **Compositional** semantics
   - paraphrasing
   - sentence similarity in applications (e.g. ordering in summarisation)
   - sentence representation learning underlies SOTA models

Modelling discourse

How?

1. **Discourse** relations
   - Classification over pairs of sentences
   - Tree-structured representations of documents
2. Learning **document representations**
   - Neural models: LSTMs, attention, HAN
   - Some later models incorporate discourse structure (ATCS)
3. **Co-reference** resolution
   - Linguistically-motivated features
   - Neural models: Lee et al (2017)

Modelling discourse

Why is this useful?

1. **Discourse** relations
   - in applications
   - e.g. summarisation: remove specific types of satellites
   - sentiment: identify contrasts in discourse
2. Learning **document representations**
   - Underlie all document classification tasks
3. **Co-reference** resolution
   - in semantics: pronouns need to be resolved
   - in applications (e.g. sentiment, summarisation)

Why does the course cover so much linguistics?

Why does the course cover so much linguistics, when all we use nowadays is machine learning anyway?

To be able to advance the state of the art you need to:

- understand the nature of the learning problem
- understand the structure of your data
- understand what patterns you might find in the data
- develop an appropriate learning algorithm for this

Understanding linguistic properties can lead to algorithmic advances in ML, e.g. the word meaning variation in context motivated the design of self-attention.

Why does the course cover so much linguistics?

Why does the course cover so much linguistics, when all we use nowadays is machine learning anyway?

To be able to advance the state of the art you need to:

- understand the nature of the learning problem
- understand the structure of your data
- understand what patterns you might find in the data
- develop an appropriate learning algorithm for this

Understanding linguistic properties can lead to algorithmic advances in ML, e.g. the word meaning variation in context motivated the design of self-attention.

Exam content

All lectures including guest lectures.

- Morphological processing
- n-gram language models
- Part-of-speech tagging
- Syntax, formal grammars and syntactic parsing
- Distributional semantics and word embeddings
- Compositional distributional semantics
- Neural sequence processing and sentence representations
- Discourse processing
- Summarisation, dialogue modelling, machine translation

You are allowed to bring a **cheat sheet** (A4) and a **calculator**.

Types of questions

- Explain a particular linguistic phenomenon and why it is challenging for particular NLP methods / applications
- Explain the strengths and limitations of a particular method
- Apply a method to a given example
- Given examples of system errors, explain why these arise
- How can one apply a method from one NLP task to solve a particular problem in another NLP task