Paper Talk Episode 4

Building Foundation Models using Transformers

Aditeya Baral

About Me

- **Cisco Webex***
  - Big Data Analytics, Webex Media Quality
  - Webex Message AI

- **Intel Research (VSG) -** Applied Research Scientist Intern
- **Center for Cloud Computing & Big Data, PESU** - UG Researcher
- **Publications**
  - AAAI MAKE 2022
  - ICNLSP 2021
  - IEEE CONIT 2021
- **Interests**
  - Representation Learning for language understanding
  - Foundation models and Multi-Modal learners
  - Low-resource or under-represented NLP

Overview of Topics

1. What is a Foundation Model and Introduction to Transfer Learning
2. Representation Learning – What, Why and How?
3. Overview of some NLU algorithms
4. CalBERT and MWP-BERT
5. Hands-on Exercise
6. Conclusion

Foundation Models

- If you can drive a BMW, can you also drive Mercedes?
- Do you know the values of the constants – speed of light, acceleration due to gravity?
- Do you know which biological component is called the powerhouse of the cell?

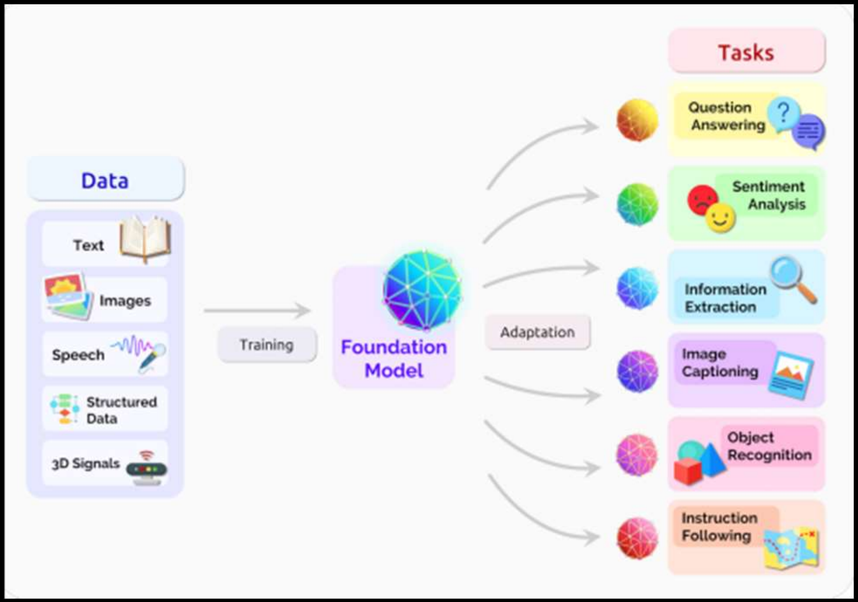
Foundation Models

- If you own and can drive a BMW, can you also drive Mercedes?
  + Can you also drive an autorickshaw? A bus? A tractor?
  + If you can drive a manual car, can you drive an automatic?

- Do you know the values of the constants – speed of light, acceleration due to gravity?
  + Do you also know the value of Plank’s constant? Newton’s Gravitation constant? 1 Mole?

- Do you know which biological component is called the powerhouse of the cell?
  + Which component is called the kitchen of the cell?

Foundation Models



- Built on *foundational* learning – just the way humans do!
  - Learn concepts A, B and C and apply them to A, B, C, as well as D, E, F...
  - Learn science → Physics, Chemistry and Biology → Quantum Physics
  - Learn numbers → add/sub/mul/div → expressions → algebra → calculus
- Trained on gigabytes of different forms of data ranging different topics
  - Ensures satisfactory performance on all tasks *without* fine-tuning
  - Compromises great performance on any single task (achieved *with* fine-tuning)
  - Models like GPT-4 are trained on academia, literature, science, legal etc
- Use foundation models as *stepping-stones*
  - Use transfer learning to improve performance on singular tasks
  - Go from a *generic* model to a *specialised* model
- Usually created on non-specific tasks
  - Mostly unsupervised or self-supervised

On the Opportunities and Risks of Foundation Models, CRFM, Stanford Institute for HAI

Transfer Learning

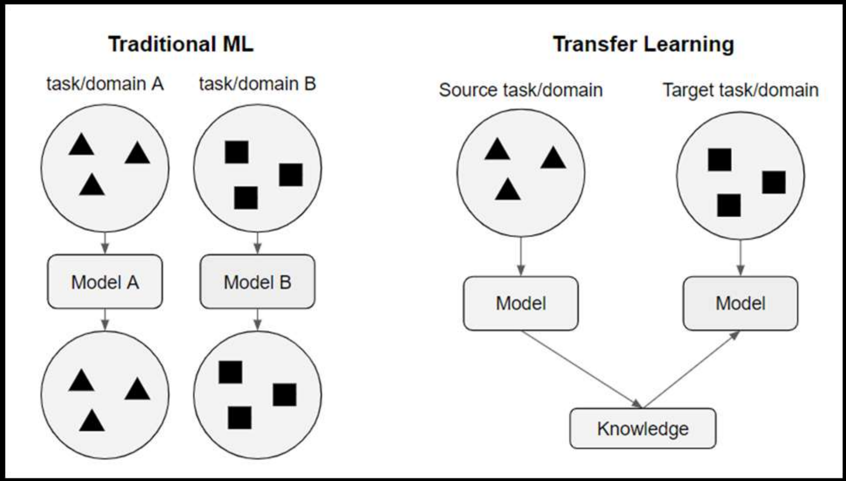
- *Application* of foundation models to different downstream tasks
  + Downstream task could be similar or dissimilar to upstream tasks
    - Study for an exam and then apply it on questions
  + A model for summarization can be fine-tuned for paraphrasing or question-answering
- Two stages
  + **Pre-training**
    - Build foundation model
    - Use large amounts of data
    - Takes more time and compute
  + **Fine-tuning**
    - Build task-specific model
    - Use small amounts of data
    - Takes less time and compute

Transfer Learning

- Example: Summarization of legal articles
  + Step 1: Pre-train on large amounts of text data
  + Step 2: Fine-tune on medium amount of legal text data
  + Step 3: Fine-tune for generating summaries

- Fine-tuning comprises 2 steps
  + **Domain adaptation** – fine-tune foundation model on the same domain as target domain
  + **Task adaptation** – fine-tune foundation model on specific tasks
  + Sometimes, both steps can be accomplished in a *single* step!

- Sometimes the same model is used for pre-training and fine-tuning
  + Freeze the embedding/initial layers



https://slds-lmu.github.io/seminar_nlp_ss20/introduction-transfer-learning-for-nlp.html

Transfer Learning

- Types/sub-fields
  + Zero-Shot (or, no transfer learning)
  + One-Shot
  + Few-Shot
  + Knowledge Distillation

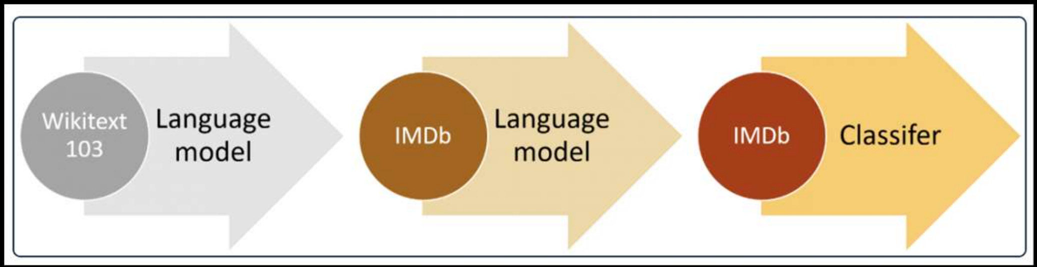
- **Advantages**
  + Requires less data to model your task, faster to adopt different tasks
    - Teaching someone who knows to drive a car to drive a bus is easier than teaching someone who cannot drive at all
  + Higher performance on targeted tasks
  + Less compute

- **Disadvantages**
  + *"Catastrophic Forgetting"*
  + Domain suitability - avoiding negative transfer

Transfer Learning and NLP

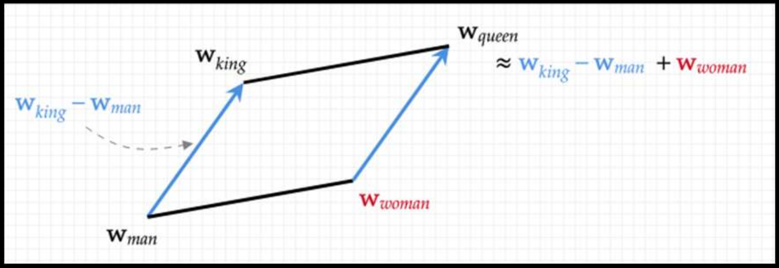
- NLP made transfer learning popular
  - Almost all modern-day NLP is based on transfer learning – ChatGPT/GPT-4, Bard, LLaMa
  - Large amount of text data is available for generic uses, but specific uses have less data
  - Build basic language understanding and then adapt to domain/task

- Main goal of transfer learning – better language understanding
  - Create models which understand better, and hence perform better
  - Tougher than modelling language/linguistic form
  - Learn representations for language – word embeddings that capture relationships and properties



Representation Learning

- Represent characters, words, sentences in a numeric form
  + Convert a word to an *n*-dimensional vector
    - Each *dimension* represents a hidden characteristic/feature of that word
  + Vector lies in a semantic space of all words in the vocabulary
    - Similar vectors lie closer (angle between them is 0)
  + Vectors have relationships with other vectors and can be operated on
    - King – man + woman = Queen (Word2Vec, 2013)



- Primitive approaches
  + Bow, Tf-Idf
- Preliminary Neural approaches
  + Word2Vec, fastText, GloVe
- Modern LM approaches
  + LSTM - ELMo, ULMFiT
  + Transformer - BERT

Representation Learning Approaches

- Bidirectional contextual learning using Language Modelling
  + A B __ D E - Masked Language Modelling (MLM)
  + _ _ C _ _ - identify correct context

- Next Sentence Prediction (NSP)
  + Learn if a sentence is followed by another
  + Sentence 1 [SEP] Sentence ?

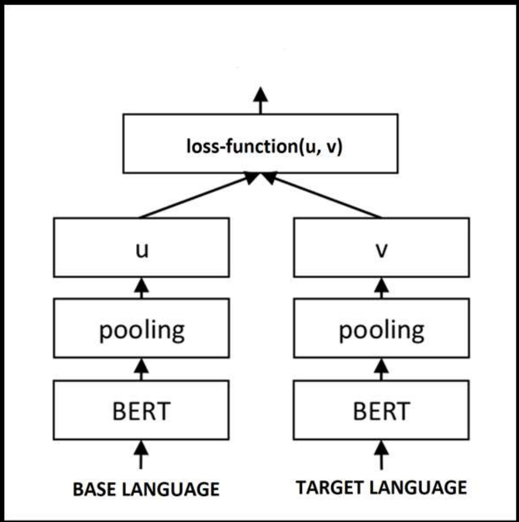
- Word2Vec was based on MLM, but it used context windows, not sentences
  + fastText did the same at the character level and then averaged embeddings for a word

- BERT uses 2 approaches for pre-training – MLM + NSP
  + BERT variants: RoBERTa (only MLM), DistilBERT (KD using pre-trained BERT), XLM-RoBERTa (multilingual)

- Current approaches for representation learning need MLM (and/or NSP) along with other specialised steps for domain adaptation

CalBERT

- Code-mixed languages are complex and are prevalent in multilingual communities
  + Multiple forms of the same word
  + Lack of abundant clean and usable data
  + Normal Transformers do not perform well on code-mixed tasks
- Two proposed techniques to learn from code-mixed data
  + Accounts for context as well as morphological mutations
  + Knowledge Distillation: “Adapt” representations in English to Hinglish
  + Pre-training: End-to-end pre-training using different tasks
    - MLM
    - NSP
    - Alignment with transliteration
    - Semantic similarity with translation and transliteration



- Applications
  + KD approach achieved SOTA on 2 code-mixed benchmarks, beating existing approaches by 9%
  + Models are task-agnostic, thus can be applied to any code-mixed task

MWPBERT

- A recent pre-training technique to learn representations of Math Word Problems (MWPs)
- MWPs are not like normal text
  + Need to learn information about operands, operators and computations
  + Representations need to also account for the *validity* of MWPs
- MWPBERT injects knowledge about numbers and solvability to existing BERT pre-training
  + MLM
  + Operand counting
  + Operand data type prediction
  + Answer data type prediction
  + Operand and Answer data type compatibility
  + Answer magnitude comparison
  + Operation prediction
  + Equation tree distance prediction
- Applications
  + Significant improvement over ordinary BERT as well as other methods for MWP solving and generation

MWP-BERT: Numeracy-Augmented Pre-training for Math Word Problem Solving

Hands-on Exercise

- Problem
  - + Ideate different tasks to learn from a structured conversation (like WhatsApp)
  - + You can use any data which WhatsApp provides
    - messages, contact info, group info
  - + Should learn accurate representations for entities, topics and develop basic language understanding

Conclusion

- Foundation models can be used to create models to tackle multiple tasks
- Using transfer learning, foundation models can be fine-tuned for different domains or tasks
  - + Domains/Tasks can be similar to pre-training stage
  - + Improves performance, requires lesser data and compute
  - + Primarily used in NLP for language understanding
- Representation learning is used to build NLU
  - + Modern methods use BERT-based architectures
  - + Leverage multiple pre-training tasks to adapt to specialised use-cases
  - + If tasks are chosen correctly, it is possible to model almost everything
    - Code-mixed languages
    - Math word problems
    - WhatsApp conversations
  - + Representations are then used for other predictive tasks

Thank You