– IN5550 –
Neural Methods in Natural Language Processing
Home Exam: Task Overview and Kick-Off

Andrey Kutuzov, Lilja Øvrelid, Roxana Pop,

Egil Rønningstad, David Samuel & Erik Velldal

University of Oslo

April 21st, 2022



Home Exam

General Idea

- Use as guiding metaphor: Preparing a scientific paper for publication.

Home Exam

General Idea

- Use as guiding metaphor: Preparing a scientific paper for publication.

Fourth IN5550 Teaching Workshop on Neural NLP (WNNLP 2022)

Home Exam

General Idea

- Use as guiding metaphor: Preparing a scientific paper for publication.

Fourth IN5550 Teaching Workshop on Neural NLP (WNNLP 2022)

Standard Process

(1) Experimentation

(2) Analysis

(3) Paper Submission

(4) Reviewing

(5) Camera-Ready Manuscript

(6) Presentation

Home Exam

General Idea

- Use as guiding metaphor: Preparing a scientific paper for publication.

Fourth IN5550 Teaching Workshop on Neural NLP (WNNLP 2022)

Standard Process

(0) Problem Statement

(1) Experimentation

(2) Analysis

(3) Paper Submission

(4) Reviewing

(5) Camera-Ready Manuscript

(6) Presentation

WNNLP 2022: Call for Papers and Important Dates



General Constraints

- Four specialized tracks: Targeted Sentiment Analysis, Word Sense Induction, Machine Translation, Semantic Parsing.
- Long papers: up to eight pages, excluding references, in ACL Rolling Review style.
- Submitted papers must be anonymous: peer reviewing is double-blind.
- Replicability: Submission backed by code repository (area chairs only).

Schedule

By April 29	Declare choice of track (and team composition)
Week 18/19	Track specific joint mentoring meeting
May 19	(Strict) Submission deadline for scientific papers
May 23–30	Reviewing period: Each student reviews two papers
June 2	Area Chairs make and announce acceptance decisions
June 7	Camera-ready manuscripts due, with requested revisions
June 9	Oral presentations and awards at the workshop



UNIVERSITY
OF OSLO

Menu ← Studies ← Courses

IN5550 - Spring 2022

Final exam

Obligatory exercises

Slides

Final exam

Between Thursday April 28 and Thursday May 19, there will be a home exam in the form of a small research project. To pass the exam, students need to submit and have accepted a scientific paper to the 4th IN5550 Teaching Workshop on Neural Natural Language Processing (WNNLP 2022).

Background

Final examination in this class takes the form of a 'home exam', i.e. a project that students can work on over a period of three weeks. Like for the exam-qualifying obligatory assignments, group work (in teams of up to three students) is encouraged. Team composition needs to be declared at the start of the exam period and, cannot be changed after April 29, 2022. Also, each team needs to decide beforehand which of the available tracks they want to research; these 'tracks' of the exam will be introduced in the lecture on April 21 (examples from previous years have included named entity recognition, negation resolution, and sentiment analysis). Further background on the tracks and supporting data and code are available through the course GitHub repo. Please announce your team composition and choice of track no later than April 29 by

Empirical (Experimental)

- Motivate architecture choice(s) and hyper-parameters;
- systematic exploration of relevant parameter space;
- comparison to reasonable baseline or previous work.

WNNLP 2022: What Makes a Good Scientific Paper?



Empirical (Experimental)

- Motivate architecture choice(s) and hyper-parameters;
- systematic exploration of relevant parameter space;
- comparison to reasonable baseline or previous work.

Replicable (Reproducible)

- Everything relevant to run and reproduce in M$ GitHub.

WNNLP 2022: What Makes a Good Scientific Paper?



Empirical (Experimental)

- Motivate architecture choice(s) and hyper-parameters;
- systematic exploration of relevant parameter space;
- comparison to reasonable baseline or previous work.

Replicable (Reproducible)

- Everything relevant to run and reproduce in M$ GitHub.

Analytical (Reflective)

- Identify and relate to previous work;
- explain choice of baseline or points of comparison;
- meaningful, precise discussion of results;
- 'negative' results can be interesting too;
- look at the data: discuss some examples:
- error analysis: identify remaining challenges.

WNNLP 2022: Programme Committee

General Chair

- Lilja Øvrelid

Track Chairs

- Targeted Sentiment Analysis: Egil Rønningstad & Erik Velldal
- Word Sense Induction: Andrey Kutuzov
- Neural Machine Translation: David Samuel
- Semantic Parsing: Roxana Pop

Peer Reviewers

- All students who have submitted a scientific paper

Track 1: Targeted Sentiment Analysis

- Sentiment Analysis:
  - identifying subjective content in text, and
  - measuring positive/negative polarity.
  - different granularities: Document-level, sentence-level, sub-sentence-level

Track 1: Targeted Sentiment Analysis

- Sentiment Analysis:
  - identifying subjective content in text, and
  - measuring positive/negative polarity.
  - different granularities: Document-level, sentence-level, sub-sentence-level
- Fine-grained sentiment analysis at the sub-sentence level
  - what is the target of sentiment?
  - what is the polarity of sentiment directed at the target?

1. Denne disken$_{POS}$ er svært stillegående
   'This disk runs very quietly'

The Norwegian Review Corpus (NoReC)

En juleklassiker er utsprungen

Album jul



Guren Hagen med Sølvguttene og Ole Edvard Antonsen
Julekveld hos oss

6 *En hjertevarm, personlig juleplate full av herlig musikalsk variasjon.*

Guren Hagen er en av våre fineste viseartister, og med *Julekveld hos oss* har han skapt ei stemningsfull juleplate med stor troverdighet. Spesielt hans egenkomponerte julesanger gjør dette til julens store stemningsspreder. Hans varme stemme og østerdalsdialekt drar deg inn hans verden, mens Sølvguttenes vakre sang og Ole Edvard Antonsens trompet gir ekstra krydder til en av årets mest vellykkede juleplater. Guren Hagen har på utsøkt vis i tittelmelodien klart å formidle hvordan han opplevde barndommens østerdalsjul slik at vi alle vil være en del av denne kosen. Samtidig gjør han Alf Prøysens «Julekveldsvisa», med hawaiigitar og lapsteel til en sann fryd. I spennet mellom disse to sangene gir han oss troen tilbake på at det finnes overraskende grep også for denne musikktradisjonen. Selv kjente travere som «Det lyser i stille grender» og «Kimer i klokker», sistnevnte i en flott duett med Lise Backstrøm, besitter en nyvunnen nærhet. Dette er blitt en vidunderlig juleplate full av musikalsk overskudd og variasjon. Musikken bærer i seg like porsjoner glede, nærhet og undring.

Svein Andersen

- Newly released dataset for fine-grained SA of Norwegian
- `https://github.com/ltgoslo/norec_fine`

	# Examples				
	Train	Dev.	Test	Total	Avg. len.
Sents.	8634	1531	1272	11437	16.8
Targets	5044	877	735	6656	2.0

Table: Number of sentences and annotated targets across the data splits.

A Fine-Grained Sentiment Dataset for Norwegian

Lilja Øvrelid, Petter Mæhlum, Jeremy Barnes, Erik Velldal
University of Oslo
Department of Informatics
{liljao,pettemae,jeremycb,erikve}@ifi.uio.no

Abstract

We introduce NoReC$_{fine}$, a dataset for fine-grained sentiment analysis in Norwegian, annotated with respect to polar expressions, targets and holders of opinion. The underlying texts are taken from a corpus of professionally authored reviews from multiple news-sources and across a wide variety of domains, including literature, games, music, products, movies and more. We here present a detailed description of this annotation effort. We provide an overview of the developed annotation guidelines, illustrated with examples, and present an analysis of inter-annotator agreement. We also report the first experimental results on the dataset, intended as a preliminary benchmark for further experiments.

Keywords: Sentiment analysis, opinion mining, Norwegian

1. Introduction

In this work, we describe the annotation of a fine-grained sentiment dataset for Norwegian, analysing opinions in terms of their polar expressions, targets, and holders. The dataset, including the annotation guidelines, is made publicly available[1] and is the first of its kind for Norwegian.

2. Related Work

Fine-grained approaches to sentiment analysis include opinion annotations as in (Wiebe et al., 2005), aspect-based sentiment (Hu and Liu, 2004), and targeted sentiment (Vo and Zhang, 2015). Whereas document- and sentence-level sentiment analysis make the simplifying assumption that all

Task specifics

- Data format: BIO (target + polarity)

sent_id = 501595-13-04

Munken	B-targ-Positive
Bistro	I-targ-Positive
er	O
en	O
hyggelig	O
nabolagsrestaurant	O
for	O
hverdagslige	O
og	O
uformelle	O
anledninger	O
.	O

Task specifics

- Data format: BIO (target + polarity)

  # sent_id = 501595-13-04

Munken	B-targ-Positive
Bistro	I-targ-Positive
er	O
en	O
hyggelig	O
nabolagsrestaurant	O
for	O
hverdagslige	O
og	O
uformelle	O
anledninger	O
.	O

- Baseline system: PyTorch pre-code for BiLSTM
- Evaluation code

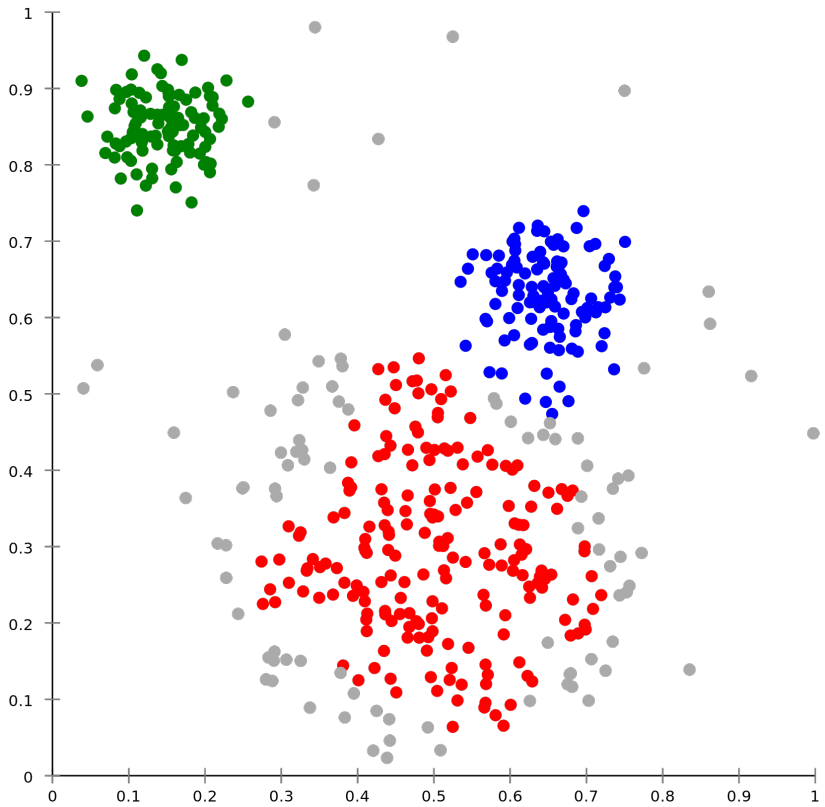
```
https:
//github.uio.no/in5550/2022/tree/master/exam/targeted_sa
```

Possible directions

1. Experiment with alternative label encoding (e.g. BIOUL)
2. Compare pipeline vs. joint prediction approaches.
3. Impact of different architectures:
   - LSTM vs. GRU vs. Transformer
   - Include character-level information
   - Depth of model (2-layer, 3-layer, etc)
4. Effect of using different pretrained word embeddings
5. Effect of using pretrained language models (ELMo, BERT)
6. Hyperparameter tuning

Track 2: Word Sense Induction

- Word Sense Induction (WSI) is discovering sense inventory from text for given input words.
- the 'unsupervised version' of word sense disambiguation (WSD):
  - automatically assigning senses to words in context from a pre-defined sense inventory;
- WSI: given a target word X and a collection of sentences where it occurs, cluster the sentences according to different senses of X.



Track 2: Word Sense Induction for Norwegian

- Human language is ambiguous at all levels.
- Syntactic ambiguity is solved by PoS taggers and dependency parsers.
- ...but words can possess different senses/meanings:

Track 2: Word Sense Induction for Norwegian

- Human language is ambiguous at all levels.
- Syntactic ambiguity is solved by PoS taggers and dependency parsers.
- ...but words can possess different senses/meanings:
  - *Among the bank[1] assets were some loans of doubtful recoverability*
  - *The river bank[2] slid by as their boat was gently carried along*
  - *Washington[1] was elected in 1788 as the first President of the United States of America*
  - *I went to Washington[2] with only the clothes on my back*
  - *Washington[3] was becoming more interested in some forms of nuclear co-operation*

Track 2: Word Sense Induction for Norwegian



- Human language is ambiguous at all levels.
- Syntactic ambiguity is solved by PoS taggers and dependency parsers.
- ...but words can possess different senses/meanings:
  - *Among the bank[1] assets were some loans of doubtful recoverability*
  - *The river bank[2] slid by as their boat was gently carried along*
  - *Washington[1] was elected in 1788 as the first President of the United States of America*
  - *I went to Washington[2] with only the clothes on my back*
  - *Washington[3] was becoming more interested in some forms of nuclear co-operation*
- Semantic processing operates with senses, not words.
- How to solve it computationally?

‘I don’t believe in word senses’



- ‘*...no single set of word senses is appropriate for different NLP applications*’
- ‘*To know what a word sense s1 is, is to know which uses of the word are part of s1, and which are not*’
- ‘*Word senses are abstractions from clusters of corpus citations*’

[Kilgarriff, 1997]

‘I don’t believe in word senses’



- ‘*...no single set of word senses is appropriate for different NLP applications*’
- ‘*To know what a word sense s1 is, is to know which uses of the word are part of s1, and which are not*’
- ‘*Word senses are abstractions from clusters of corpus citations*’

[Kilgarriff, 1997]

All models are wrong, but some are useful.

Track 2: Word Sense Induction for Norwegian

- One can infer sense inventories from corpora automatically.
- No pre-defined sense inventories.
- INPUT is a corpus, OUTPUT is sense sets for the target words, given the corpus.

Track 2: Word Sense Induction for Norwegian

- One can infer sense inventories from corpora automatically.
- No pre-defined sense inventories.
- INPUT is a corpus, OUTPUT is sense sets for the target words, given the corpus.
- The word sense induction (WSI) task is:
  1. infer a sense inventory for each target word type
  2. this includes determining the number of senses (lumping / splitting)
  3. assign a sense (or senses) from the inferred inventory to each target word token (usage / instance).

Track 2: Word Sense Induction for Norwegian

- One can infer sense inventories from corpora automatically.
- No pre-defined sense inventories.
- INPUT is a corpus, OUTPUT is sense sets for the target words, given the corpus.
- The word sense induction (WSI) task is:
  1. infer a sense inventory for each target word type
  2. this includes determining the number of senses (lumping / splitting)
  3. assign a sense (or senses) from the inferred inventory to each target word token (usage / instance).

Word sense induction is what lexicographers do when compiling dictionaries (except coining definitions).

Evaluation

- compare assignment of induced senses to word usages and the gold annotation;
- this means comparing two different groupings / clusterings.

Evaluation

- compare assignment of induced senses to word usages and the gold annotation;
- this means comparing two different groupings / clusterings.
- Clustering evaluation metrics: Adjusted Rand Index (ARI).
- ARI is the chance-corrected version of Rand index:
  - $Rand = \frac{\#AgreeingPairs}{\#AgreeingPairs+\#DisagreeingPairs}$
  - Probability that two clusterings will agree on a random pair.
  - Yields 0 for a random clustering, 1 for the ideal clustering.
  - Easily interpretable
- Provided in `scikit-learn`.

WSI with word embeddings

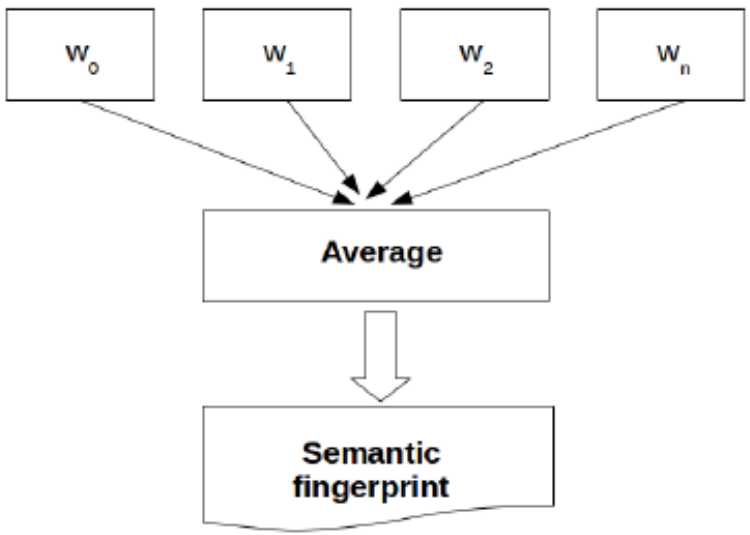
1. Each word in the context sentence is mapped to its vector in a pre-trained word embedding model;

WSI with word embeddings

1. Each word in the context sentence is mapped to its vector in a pre-trained word embedding model;
2. Ambiguous query word itself is removed;

WSI with word embeddings

1. Each word in the context sentence is mapped to its vector in a pre-trained word embedding model;
2. Ambiguous query word itself is removed;
3. For each context utterance, the average of all words' vectors is computed.
4. This dense vector is the semantic representation of the context utterance.



Context clustering

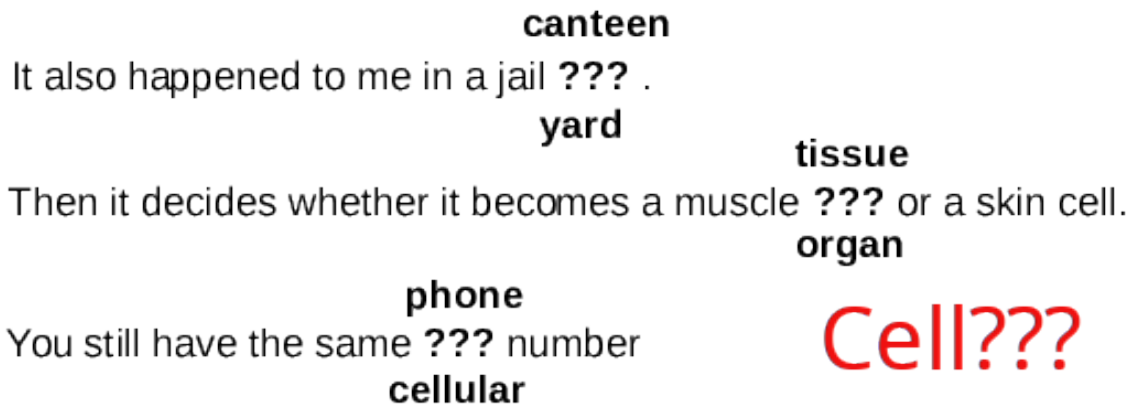
- Any clustering algorithm can be used to group contexts into sense clusters:
  - *K-means*
  - *agglomerative clustering*
- But if the number of senses (clusters) for each query word is unknown, the algorithm should be able to induce it from the data.
- *Affinity Propagation* [Frey and Dueck, 2007] is often used in this case.
- It detects the most probable number of clusters and provides the clustering itself.

Context clustering

- Any clustering algorithm can be used to group contexts into sense clusters:
  - *K-means*
  - *agglomerative clustering*
- But if the number of senses (clusters) for each query word is unknown, the algorithm should be able to induce it from the data.
- *Affinity Propagation* [Frey and Dueck, 2007] is often used in this case.
- It detects the most probable number of clusters and provides the clustering itself.

It is also possible to map averaged embeddings to sense representations derived from ego graphs of word's nearest neighbors [Logacheva et al., 2020].

WSI with contextualized language models



Lexical substitutes

- LM predicts a distribution of possible substitutes for a target word.
- senses are induced by clustering the resulting substitute vectors. [Başkaya et al., 2013]
- dynamic symmetric patterns (*'X and...?'*) additionally improve the performance. [Amrami and Goldberg, 2018]

It can be even simpler

- Simply cluster contextualized vectors of target words
- no need to average static vectors of all words in context

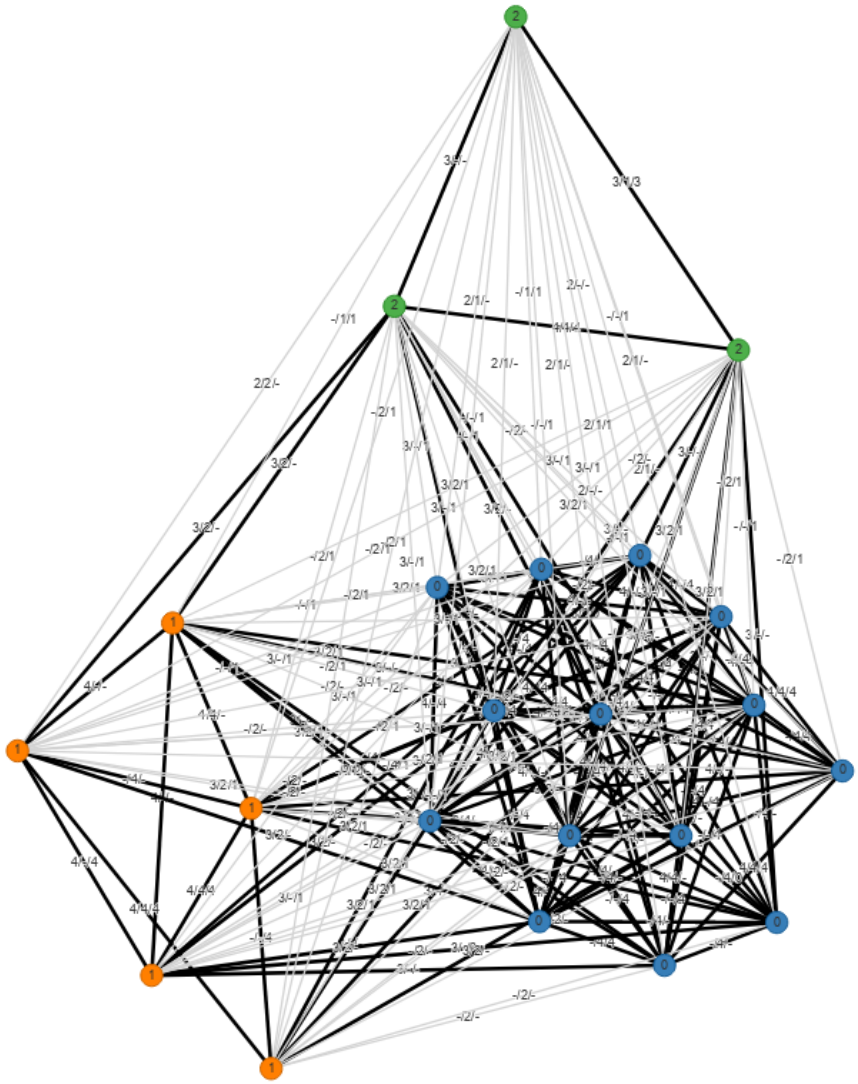
WSI for Norwegian

- There are no well established WSI datasets for Norwegian.
- But there is a recently published *NorDiaChange* [Kutuzov et al., 2022] dataset
- Originally designed for historical semantic change detection...

WSI for Norwegian

- There are no well established WSI datasets for Norwegian.
- But there is a recently published *NorDiaChange* [Kutuzov et al., 2022] dataset
- Originally designed for historical semantic change detection...
- ...contains WSI information by design.
- You will convert it to a suitable format...
- ...and evaluate SOTA WSI approaches on Norwegian data!

NorDiaChange: clustering word senses



Word usage graph for the word '*oppvarming*'.

Workflow

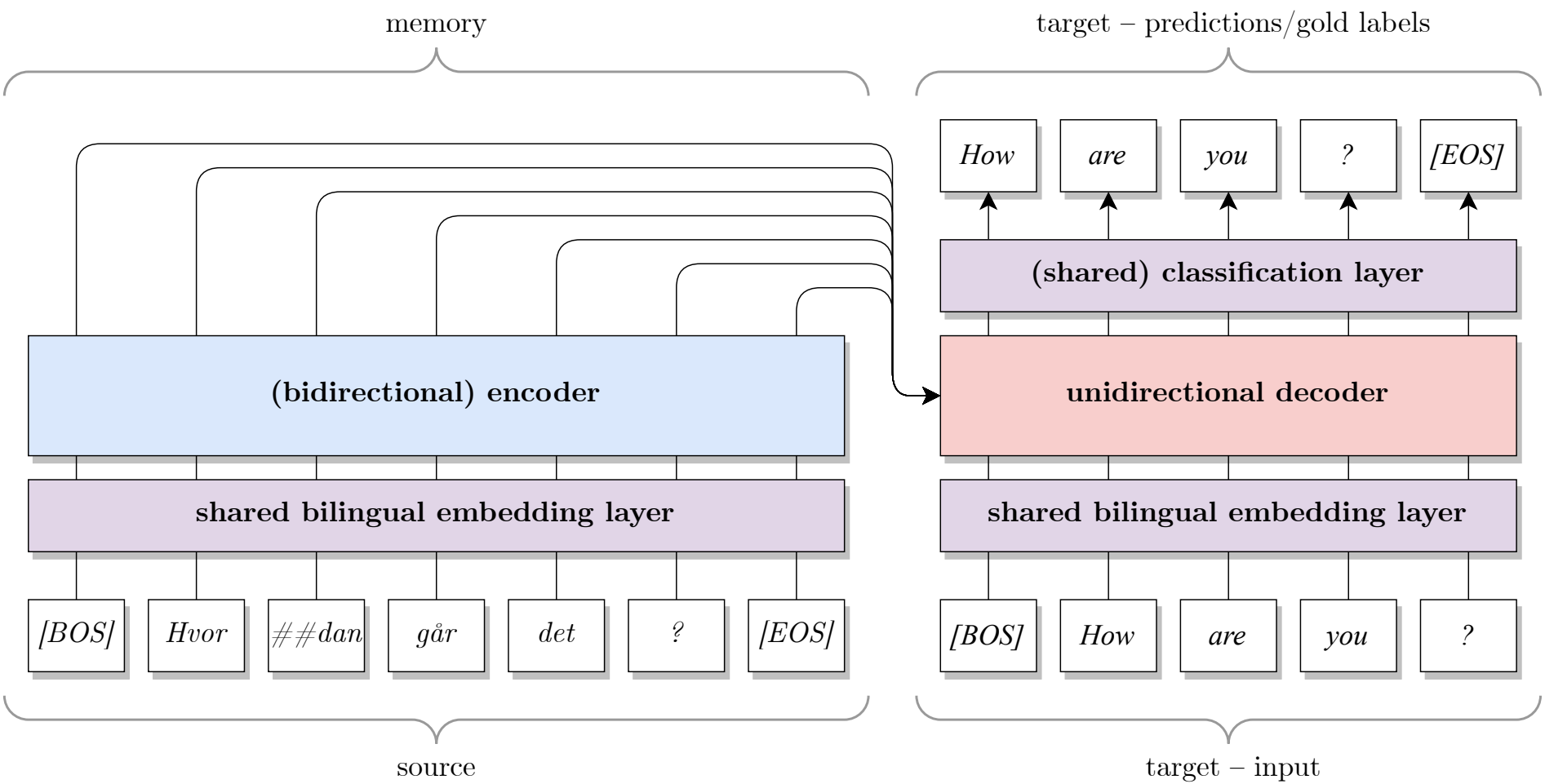
1. Convert *NorDiaChange* to a format usable for WSI evaluation.
2. Use the `egvi` algorithm from [Logacheva et al., 2020] as the baseline.
3. Reproduce it and evaluate the results on *NorDiaChange*.
4. It is possible to re-implement `egvi` in full, or to re-use the sense inventories the authors made available.
5. Reproduce a number of methods employing contextualized language models:
   - [Amrami and Goldberg, 2018]
   - [Amrami and Goldberg, 2019]
   - [Arefyev et al., 2020]
   - You are welcome to find more!
6. Evaluate them as well.
7. Play with hyperparameters.
8. Discuss the results.

Possible directions

1. What layers of language model to use?
2. Is it possible to fine-tune the models for the WSI task or we should stick to frozen models?
3. Fixed or dynamic number of clusters?
4. If dynamic, how to find the correct number?
5. Does lemmatizing the data help?
6. What about multilingual language models (XLM-R, etc)?
7. ...

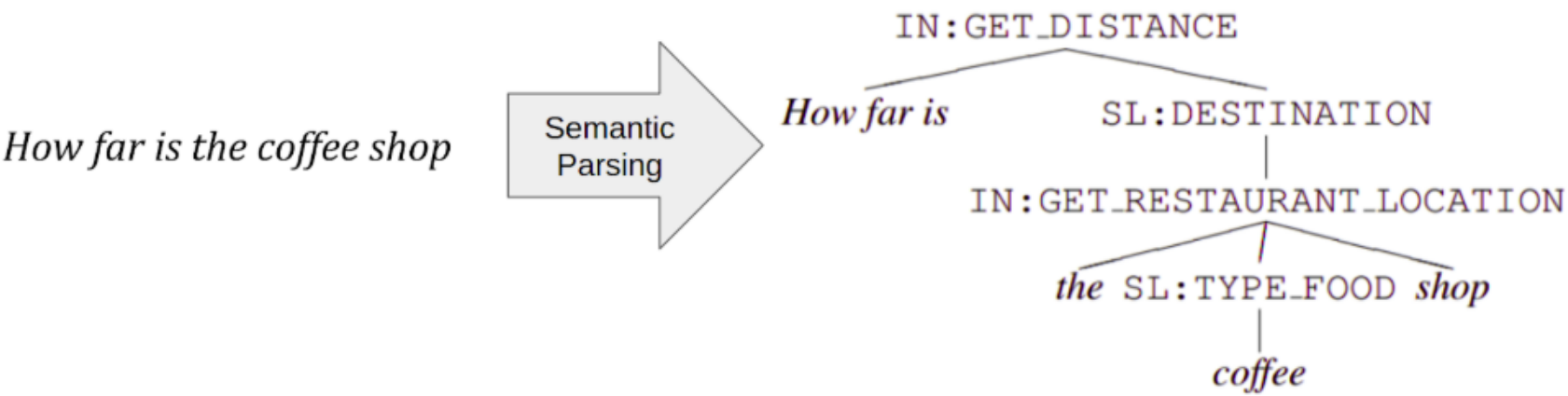
Track 3: Neural Machine Translation

See more in the detailed task description: `https://github.uio.no/in5550/2022/blob/main/exam/nmt/description_nmt.pdf`



Track 4: Semantic Parsing

- Semantic Parsing
  - Task: extract meaning representation from natural language
  - There are many such meaning representations
  - **TOP**: Task Oriented Parsing representation



TOP dataset

- Dataset of 44K instances available at `http://fb.me/semanticparsingdialog`

Split	Instances
Train	31K
Dev	4.5K
Test	9K

Table: TOP number of instances

Average input length	Average output length
Average input length	9
Average output length	17
Max input length (train)	56
Max output length (train)	64

Table: TOP statistics

TOP dataset

```
Input:  How far is the coffe shop
Output: [IN:GET_DISTANCE How far is [SL:DESTINATION
        [IN:GET_LOCATION [SL:CATEGORY_LOCATION the coffee
        shop ] ] ] ]

Input:  what are the roads looking like this morning
Output: [IN:GET_INFO_TRAFFIC what are the roads looking
        like [SL:DATE_TIME this morning ] ]

Input:  Any events in our neighborhood with food and music
Output: [IN:GET_EVENT Any events in [SL:LOCATION
        [IN:GET_LOCATION_HOME [SL:CONTACT our ] neighborhood
        ] ] with [SL:CATEGORY_EVENT food and music ] ]
```

Possible directions

- Two main directions
  - Transition-based parsing
  - Standard sequence to sequence

Possible directions

- Two main directions
  - **Transition-based parsing**
    - Goal: predict a sequence of actions that when applied to the input sentence generate the output representation.

Possible directions

- Two main directions
  - **Transition-based parsing**
    - Goal: predict a sequence of actions that when applied to the input sentence generate the output representation.
    - Form of structured prediction.

Possible directions

- Two main directions
  - **Transition-based parsing**
    - Goal: predict a sequence of actions that when applied to the input sentence generate the output representation.
    - Form of structured prediction.
    - The output will be valid.

Possible directions

- Two main directions
  - **Transition-based parsing**
    - Goal: predict a sequence of actions that when applied to the input sentence generate the output representation.
    - Form of structured prediction.
    - The output will be valid.
    - First parser on TOP was a transition-based parser

Possible directions

- Two main directions
  - **Transition-based parsing**
    - Goal: predict a sequence of actions that when applied to the input sentence generate the output representation.
    - Form of structured prediction.
    - The output will be valid.
    - First parser on TOP was a transition-based parser
    - Most work on TOP since has **not** been based on transition-based parsing

Possible directions

- Two main directions
  - **Transition-based parsing**
    - Goal: predict a sequence of actions that when applied to the input sentence generate the output representation.
    - Form of structured prediction.
    - The output will be valid.
    - First parser on TOP was a transition-based parser
    - Most work on TOP since has **not** been based on transition-based parsing
    - But, there are semantic parsing tasks where these types of systems are SOTA

Possible directions

- Two main directions
  - **Transition-based parsing**
    - Goal: predict a sequence of actions that when applied to the input sentence generate the output representation.
    - Form of structured prediction.
    - The output will be valid.
    - First parser on TOP was a transition-based parser
    - Most work on TOP since has **not** been based on transition-based parsing
    - But, there are semantic parsing tasks where these types of systems are SOTA
  - Standard sequence to sequence

Possible directions

- Two main directions
  - Transition-based parsing
  - **Standard sequence to sequence**
    - Predict the linearized tree.

Possible directions

- Two main directions
  - Transition-based parsing
  - **Standard sequence to sequence**
    - Predict the linearized tree.
    - Input: sequence, output: sequence (sequence to sequence architecture)

Possible directions

- Two main directions
  - Transition-based parsing
  - **Standard sequence to sequence**
    - Predict the linearized tree.
    - Input: sequence, output: sequence (sequence to sequence architecture)
    - Most work on TOP follows this approach

Possible directions

- Two main directions
  - Transition-based parsing
  - **Standard sequence to sequence**
    - Predict the linearized tree.
    - Input: sequence, output: sequence (sequence to sequence architecture)
    - Most work on TOP follows this approach
    - The systems with the best results use **pointer generator networks**

Possible directions

- Regardless of the type of system you can use (for the encoder and/or decoder):
  - LSTM-based models
  - transformer-based models
  - pretrained language models like BERT (encoder), BART (also for the decoder) etc.

Support and expectations

- You are provided with:
  - Three datasets (you are only required to work with one of them)
  - Baseline (from data loading to generating textual output)
  - Several references structured according to the directions that you might choose
  - More resources on demand
- You are expected to:
  - Do a literature review (starting with the provided papers)
  - Implement one or more parsing systems and experiment with different configurations
    - You have a lot of freedom here
    - You can reproduce some of the works in the literature or come up with your own architecture
  - Write the report

References I


Amrami, A. and Goldberg, Y. (2018).
Word sense induction with neural biLM and symmetric patterns.
In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 4860–4867, Brussels, Belgium. Association for Computational Linguistics.


Amrami, A. and Goldberg, Y. (2019).
Towards better substitution-based word sense induction.
arXiv preprint arXiv:1905.12598.


Arefyev, N., Sheludko, B., Podolskiy, A., and Panchenko, A. (2020).
Always keep your target in mind: Studying semantics and improving performance of neural lexical substitution.
In Proceedings of the 28th International Conference on Computational Linguistics, pages 1242–1255, Barcelona, Spain (Online). International Committee on Computational Linguistics.

References II


Başkaya, O., Sert, E., Cirik, V., and Yuret, D. (2013).
AI-KU: Using substitute vectors and co-occurrence modeling for word sense induction and disambiguation.
In Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), pages 300–306, Atlanta, Georgia, USA. Association for Computational Linguistics.


Frey, B. and Dueck, D. (2007).
Clustering by passing messages between data points.
Science, 315(5814):972–976.


Kilgarriff, A. (1997).
I don't believe in word senses.
Computers and the Humanities, 31(2):91–113.

References III



Kutuzov, A., Touileb, S., Mæhlum, P., Enstad, T. R., and Wittemann, A. (2022).
NorDiaChange: Diachronic semantic change dataset for Norwegian.
In Proceedings of the 13th Language Resources and Evaluation Conference (LREC). European Language Resources Association.



Logacheva, V., Teslenko, D., Shelmanov, A., Remus, S., Ustalov, D., Kutuzov, A., Artemova, E., Biemann, C., Ponzetto, S. P., and Panchenko, A. (2020).
Word sense disambiguation for 158 languages using word embeddings only.
In Proceedings of the 12th Language Resources and Evaluation Conference, pages 5943–5952, Marseille, France. European Language Resources Association.