Een vraagstuk kiezen
Kies een vraagstuk waar je deep learning nodig hebt. Afbeeldingen, tekst en timeseries zijn eigenlijk altijd wel goed. Tekst-data vraagt bv vaak veel meer preprocessing: daar houd ik rekening mee tijdens het beoordelen! Als je data meer werk vraagt, krijgt dat onderdeel meer gewicht voor de eindbeoordeling.

Hele simpele datasets (bv te klein, met weinig features) zijn vaak beter geschikt voor klassieke modellen dan voor deep learning, en dus niet geschikt voor deze opdracht.

Om je een idee te geven van een aantal mogelijke opties:

- process je eigen whatsapp text data, en maak bijvoorbeeld een model dat in staat is om te voorspellen van wie een bericht komt. Of label je tekstberichten in periodes, en kijk of je de periode kunt voorspellen. Of label een aantal soorten chats (bv familie / werk, of dates / vrienden) en kijk of je onbekende chats kunt labelen.
- Maak een siamees netwerk dat echte en valse handtekeningen kan onderscheiden.
- Voorspel “kopen/verkopen” momenten voor crypotcurrency timeseries
- Voorspellen van files , bijvoorbeeld in combinatie met het weer
- Er is uberhaupt veel data beschikbaar op https://data.overheid.nl/datasets
- maar ook tensorflow heeft datasets die leuk genoeg zijn (hoewel sommige dataset natuurlijk te voor de hand liggend zijn, maar veel datasets hebben een paper en beschrijving en zijn best interessant om te onderzoeken, of om net iets anders mee te doen dan in de paper staat.

Modelleren I
Hoe zie je op grote lijnen je model voor je?
Zie je het als een classificatie, regressie, clustering, timeseries probleem?
Hoeveel data heb je? Hoe goed zijn je labels? Hoeveel ruis / anomalies verwacht je?
Welke relaties verwacht je te vinden? Welke domeinkennis speelt een rol? Verwacht je lineaire relaties, complexe relaties?

Schets in grote lijnen het model dat je verwacht.
Wat voor loss functie heb je nodig? Wat voor metrics?
Wat verwacht je te kunnen voorspellen?

Als je dataset al besproken is elders (bijvoorbeeld in een paper) neem dat dan mee. Let op -> op kaggle staat echt heel veel code van hele lage kwaliteit. Ik vind bijna alles daar teleurstellend slecht. Kopieren van kaggle code doet je werk meestal meer slecht dan goed!

Data preprocessing / verkenning
Heb je al preprocessed data, of moet je nog preprocessing doen?
Zeg iets over: balanced / unbalanced; de noodzaak tot schalen; anomalies.
Kun je met behulp van plots al correlaties zien? (soms leent data zich daarvoor, soms niet)

Moet je je model bijstellen obv de voorverkenning van je data?
maak een train-validatie-test split, gebruik datagenerators

Modelleren II

Leg uit wat je model doet. Dat mag in een enkele regel per stap (dus, bv, beschrijf wat je word embedding doet in 1 of 2 regels, wat een LSTM doet in 1 of 2 regels, wat je dense layer doet in 1 of 2 regels). Als je de stof beheerst, kun je dit to-the-point!

Leg uit welke keuzes je maakt in je model, en waarom.
Zeg iets over de ordes van grootte van je model.

Kies een baseline die iets simpeler is dan je verwacht nodig te hebben.

Tunen
Beschrijf de verkenning van je model.
Ook als dingen afvallen, beschrijf waarom ze afvallen.
Analyseer de performance, en kijk vooral naar meer dan alleen je accuracy! Dus: analyseer trainsnelheid, zeg iets over learning rate, zeg iets over overfitten.

Laat zien dat je onderzoek doet, en intereert. Dus: je maakt een model, analyseert performance, verzint een volgende stap.

Performance
Zorg dat je de performance goed meet.
Gebruik en interpreteer de juiste middelen, afhankelijk van het soort probleem (classificatie, regressie, timeseries, etc)

Conclusie
Trek een eindconclusie.
Een tegenvallende performance is NIET erg.
Wat wel erg is, is als ik niet kan volgen dat je onderzoek doet, en wat je gedachtengang is mbt de keuzes die je maakt.

Extras
Alles wat we niet in de les hebben behandeld, valt onder extras.
Dus dingen als dimensionality reduction (hoofdstuk 8), unsupervised learning (hoofdstuk 9), autoencoders (hoofdstuk 17) of dingen als siamese networks. Maar ook het kiezen van een uitdagende dataset valt daaronder. Als je twijfelt, bespreek dat dan met de docent.

Iedereen gebruikt code van elkaar; bijvoorbeeld de tensorflow tutorials zijn een uitstekend startpunt. Het is niet erg om te doen, maar verwijs daar dan wel even naar! Een comment is genoeg, met een verwijzing naar de url waar de code vandaan komt. Verwijzen maakt het verschil tussen plagiaat en goed werk! Ik vind het belangrijker dat je in je analyse en bespreking laat zien, dat je begrijpt wat code doet en dat je hem kunt aanpassen naar jouw situatie.