1 Slide:

Good health to all present

My name is Kachalov Ilya. I am first year master degree students at NUST MISIS

I want to represent to you my start-up project about a Cloze-driven Pretraining of Self-attention Networks.

At first we would talk about problem statements and my solution.

After that part I will tell about target audience.

My presentation will take about ten minutes.

2 Slide:

Language model pretraining has recently been  
shown to provide significant performance gains  
for a range of challenging language-understanding problems. wever, existing  
work has either used unidirectional (left-to-right)  
language models (LMs) or bi-directional (both left-to-right and right-to-left)  
LMs (BiLMs) where each direction is trained with  
an independent loss function. My company in my person want show that even larger performance gains are possible by jointly pretraining  
both directions of a large language-model-inspired  
self-attention cloze model.

My bi-directional transformer architecture predicts every token in the training data (Figure 1).  
I achieve this by introducing a cloze-style training objective where the model must predict the  
center word given left-to-right and right-to-left  
context representations. My model separately  
computes both forward and backward states with a masked self-attention architecture, that closely  
resembles a language model. At the top of the network, the forward and backward states are combined to jointly predict the center word. This approach allows us to consider both contexts when  
predicting words and to incur loss for every word  
in the training set, if the model does not assign it  
high likelihood.

3 Slide:

I use the following approach to fine-tune the  
pretrained two tower model to specific downstream tasks. Using the model of two self-attentional towers each consisting of N stacked  
blocks: the forward tower operates left-to-right  
and the backward tower operates in the opposite  
direction. To predict a token, I combine the  
representations of the two towers, as described in  
more detail below, taking care that neither representation contains information about the current  
target token.

**Classification and regression tasks.** For single sentence classification tasks, I consider the  
language model outputs for the boundary tokens  
which I add before the start and end  
of each sentence. The output of the projection  
is softmax-normalized and the model is optimized  
with cross-entropy for classification tasks.

**Structured prediction tasks.** For named entity  
recognition and parsing I use task-specific architectures which I fine-tune together with the language model but with different learning rate. The  
architectures are detailed in the respective results  
sections. The input to the architectures are the  
output representations of the pretrained language  
model.

**No Masking.** For fine-tuning, I found it beneficial to remove masking of the current token in  
the final layer that pools the output of the two towers. It is important to have access to information  
about the token to be classified for token level classification tasks such as NER but I also found  
this to perform better for sentence classification  
tasks. In practice, I completely disable masking  
in the combination layer so that it operates over  
all forward and backward states. However, disabling masking below the combination layer does  
not perform well

4 Slide:

Results of the algorithm in question are presented on slide 3. I investigate how much pre-training benefits  
from larger training corpora. And how the domain of the data influences end-task performance.  
Figure 3 shows that more training data can significantly increase accuracy. I train all models  
with the exact same hyper-parameter settings on Common Crawl data using the CNN base architecture for 600K updates.

In conclusion, I want to say that mine start - up will be of interest to companies using solutions characteristic of machine learning