Human-level play in the game of *Diplomacy* by combining language models with strategic reasoning

Meta Fundamental AI Research Diplomacy Team (FAIR), Bakhtin, Brown, Dinan, Farina, Flaherty, Fried, Goff, Gray, Hu, Jacob, Komeili, Konath, Kwon, Lerer, Lewis, Miller, Mitts, Renduchintala, Roller, Rowe, Shi, Spisak, Wei, Wu, Zhang, Zijlstra

Table of Contents

1. Background
2. Overview of Cicero
3. Method
4. Discussion and Direction for Future Research
5. Q and A

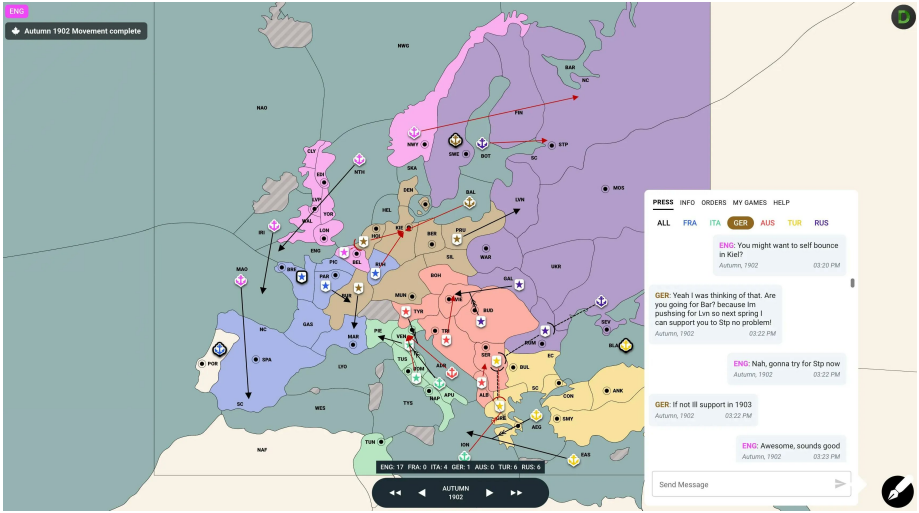
AI and Natural Language

- Long-term goal for AI to build agents that can plan, coordinate and negotiate with humans in natural language
  - Progress in imitating human language, but effective negotiation agents must go beyond
    - by understanding the beliefs, goals, and intentions of their partner
    - planning joint actions
    - and communicating these proposals

Human-level performance in *Diplomacy*

- Cicero - the first AI to achieve human-level performance in *Diplomacy*
  - 7 players conduct private natural language negotiations
    - to both cooperate and compete with each other
  - prior successes for multi-agent AI in purely adversarial environments
    - e.g. Chess, Go, and Poker
    - communication has no value
  - *Diplomacy* serves as a challenging benchmark

Snapshot of Cicero in its natural habitat



Quick Summary

- Cicero combines a controllable dialogue module with a strategic reasoning engine
  - At each point in the game, Cicero models how the other players are likely to act based on the game state and their conversations
  - Then plans how the players can coordinate to their mutual benefit
  - And maps these plans into natural language messages
- In 40 games of Diplomacy in an online league of human players between August and October:
  - Cicero clocked 72 hours of play
  - Involving sending 5277 messages
  - And ranked in the top 10% of participants

Challenges of human-AI cooperation in Diplomacy

Prior AI breakthroughs in games in two-player zero-sum (2p0s) settings

- including Chess, Go, heads-up poker, and Starcraft
- certain RL algorithms that learn by playing against themselves (self-play) will converge to an unbeatable policy in expectation in balanced games
  - Not the case in games involving cooperation without human data
    - ★ even with infinite compute and model capacity
    - ★ may converge to an incompatible policy with human norms
    - ★ self-play algorithm performed poorly even in dialogue-free versions of diplomacy

Maintaining human-interpretable communication

- The challenge of maintaining human interpretable communication is significant in *Diplomacy*
  - the AI agent dealt with 292 message per game on average
  - messages often involved precise plans
- Each message must be "grounded in" (i.e. contextually appropriate and consistent with) lengthy dialogue histories, game states, and goals
  - If inaccurately grounded
    - ⋆ the agent may be asked to explain its errors
    - ⋆ humans may cooperate with others instead

Building trust

- Success in *Diplomacy* hinges on building trust with others in an environment that encourages players not to trust anyone
  - actions occur simultaneously after non-binding, private negotiations
- an agent must account for the possibility of being betrayed, or other players may doubt the agent, therefore the agent needs
  - the ability to reason about the beliefs, goals and intentions of others, and
  - the ability to persuade and build relationships through dialogues

The game of *Diplomacy*

- The objective: to control supply centres (SCs) on a map
- The game ends when:
  - a player wins by controlling a majority of SCs, or
  - all players agree to a draw, or
  - a turn limit is reached
    - ⋆ scores are determined based on the number of controlled SCs
- All players engage in private pairwise free-form dialogue during a negotiation period each turn
  - all players then simultaneously choose an action comprised of one order per unit they control
  - a unit may support other units (can belong to other players)
    - ⋆ the basis for much of the negotiation in *Diplomacy*

Table of Contents

1 Background

2 Overview of Cicero

3 Method

4 Discussion and Direction for Future Research

5 Q and A

Overview of Cicero

Cicero combines a controllable dialogue module with a strategic reasoning engine

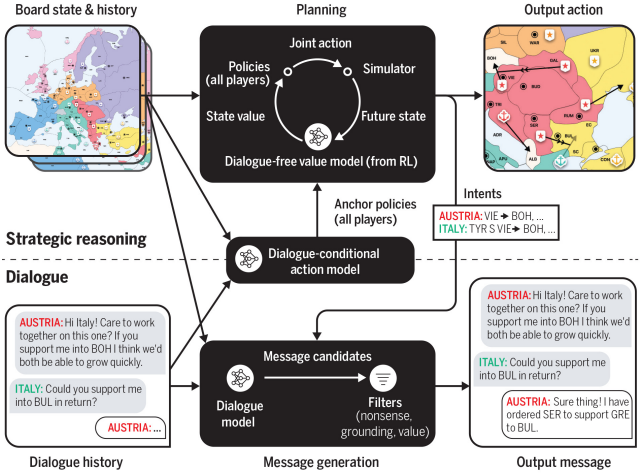


Table of Contents

1 Background

2 Overview of Cicero

3 Method

4 Discussion and Direction for Future Research

5 Q and A

Data

- Dataset of 125,261 games of Diplomacy played at webDiplomacy.net
  - 40,408 contained dialogue, total of 12,901,662 messages exchanged
  - player accounts de-identified
  - dataset hereafter referred to as WebDiplomacy

Imitation dialogue model

- Base model: R2C2
  - a 2.7B parameter Transformer-based encoder-decoder model
  - pre-trained on text from the Internet using BART de-noising objective
- further trained on WebDiplomacy via standard MLE. With a dataset $\mathcal{D} = \left\{ \left[\mathbf{x}^{(i)}, \mathbf{y}^{(i)} \right] \right\}$
  - the model is trained to predict a dialogue message $\mathbf{y}^{(i)}$ from player A to player B at time t, given all of the following represented as text $\mathbf{x}^{(i)}$:
    - ★ diaglogue history
    - ★ game state and action history
    - ★ player rating
    - ★ game and message metadata
    - ★ intents

Controllable dialogue model via intents

- Standard language modelling merely imitates messages from the dataset, not to outperform
  - to go beyond, dialogue was made controllable by generating messages conditioned on a plan
    - ⋆ specified by strategic reasoning modules
    - ⋆ resulting in higher quality messages
  - a message has intent $\mathbf{z}$ if $\mathbf{z}$ is the most likely set of actions the sender and recipient will take
    - ⋆ for both the current and several future turns
    - ⋆ if no further dialogue occurs
  - techniques were developed to annotate every message in the training set with a set of actions
    - ⋆ the distribution $p_\theta\left[\mathbf{y}^{(i)}|\mathbf{x}^{(i)}, \mathbf{z}^{(i)}\right]$ where $\mathbf{z}^{(i)}$ represents the intent for datapoint $\left[\mathbf{x}^{(i)}, \mathbf{y}^{(i)}\right]$

The effects of intents on Cicero's dialogue

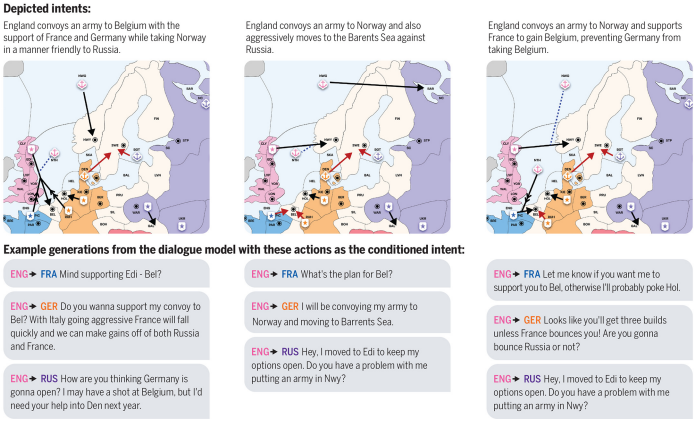


Figure: Three of Cicerro(ENG)'s possible intents and corresponding generated dialogue.

Training and inference for intent-controlled dialogue

- the goal was for the proposed actions $\mathbf{z}^{(i)}$
  - to closed reflect the content of a message $\mathbf{y}^{(i)}$
- naïve way: use actual actions played by the sender and recipient
  - but actions do not necessarily reflect the content
    1. the message is not honest
    2. subsequent messages change the sender's plans
  - issues resolved by:
    1. training on a 'truthful' subset of the dataset
    2. intent model seeing dialogue history up to the message in question

Training and inference for intent-controlled dialogue

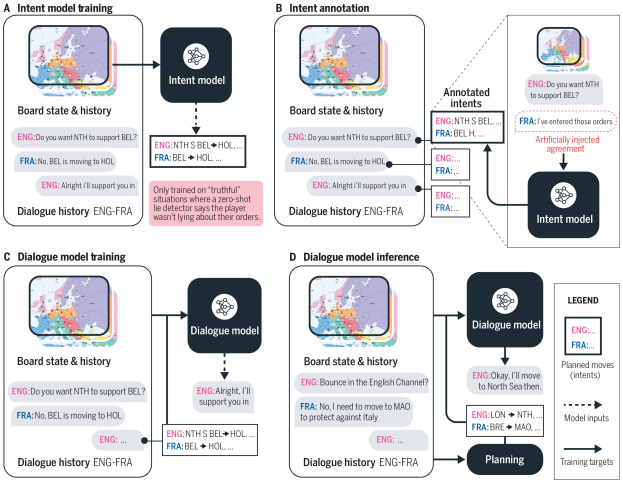


Figure: To select intents during play, Cicero uses the strategic reasoning module.

Agent intent action and recipient intent action for the current turn

- Agent intent action
  - Dialogue conditioned on the action that Cicero intends to play for the current turn
    - ⋆ maximises honesty and ability to coordinate
    - ⋆ but risks information leakage being exploited
- Recipient intent action
  - Cicero considers recipient actions with high likelihood, which requires either
    - ⋆ an action is deemed beneficial for the recipient and/or
    - ⋆ action likely to be played given dialogue
  - recipient action with the highest EV for itself is selected

Dialogue model results

	DIALOGUE QUALITY RATINGS (%)			
	Consistent with state	**Consistent with plan**	**High quality**	**Perplexity**
Language model	61.90	76.19	20.64	8.02
+ game state grounding	84.13	83.33	29.37	7.94
+ intent grounding (CICERO)	87.30	92.86	37.30	7.70

Figure: The performance of the dialogue model was compared to a baseline.

Strategic reasoning

- the strategic reasoning module predicts other players' policies
  - for the current turn
  - based on board state and dialogue
- chooses a policy for itself for the current turn
  - responds optimally to other players' predicted policies
- common approach in cooperative games to model other players' policies via supervised learning
  - aka **Behavioural Cloning** (BC)
  - but pure BC is brittle
    - ⋆ spurious correlations between dialogue and actions may be learned
    - ⋆ Cicero used variants of piKL to model

piKL: KL-regularised planning

- an iterative algorithm that predicts player policies by treating each turn in *Diplomacy* as its own subgame
  - each player i simultaneously chooses an action $a_i \Rightarrow$ jointed action $a = (a_1, ..., a_n)$
  - each player i receives a reward $u_i(a)$ determined by a value function u_i
- assumes player i seeks a policy π_i to maximise

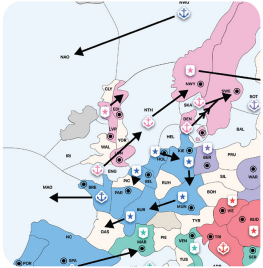
$$U_i(\pi_i, \pi_{-i}) = u_i(\pi_i, \pi_{-i}) - \lambda D_{KL}(\pi_i || \tau_i)$$

  - π_{-i} the policies of all players other than i
  - $u_i(\pi_i, \pi_{-i})$ the EV of π_i given π_{-i}
  - τ_i the BC/anchor policy
  - maxmises u_i and minimises KL divergence

Dialogue conditional planning

England agrees:

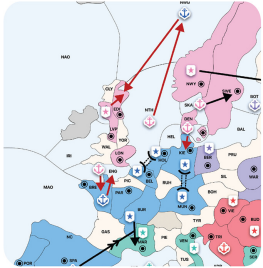
ENG➔ FRA Yes! I will move out of ENG if you head back to NAO.



Cicero predicts England will retreat from ENG to NTH 85% of the time, backs off its own fleet to NAO as agreed, and begins to move armies away from the coast.

England is hostile:

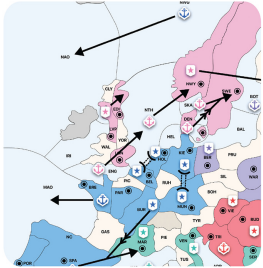
ENG➔ FRA You've been fighting me all game. Sorry, I can't trust that you won't stab me.



Cicero does not back off its fleet but rather attacks EDI with it, and leaves its armies at the coast to defend against an attack from England, predicting that England will attack about 90% of the time.

England tries to take advantage of Cicero:

ENG➔ FRA Yes! I'll leave ENG if you move KIE -> MUN and HOL -> BEL.



Strategic planning rejects the possibility of vacating KIE and HOL, because it would make Cicero too vulnerable. Cicero backs off its fleet to NAO but keeps armies at the coast to defend.

Figure: Cicero(FRA) has just messaged ENG 'Do you want to call this fight off? I can let you focus on Russia and I can focus on Italy.'

Message filtering

- Neural language models suffer from contradictions, inconsistencies and a tendency to 'hallucinate', or generate factually incorrect information
- all exhibited in *Diplomacy*
  - authors approached this problem by filtering using a series of classifiers and checks common issues
  - filters include
    - Discriminating between human text and counterfactuals
    - intent correspondence filters
    - value based filtering

Cicero in anonymous human play

- Participated anonymously in 40 games on webDiplomacy.net
  - 5 minute negotiation turns
  - allowed games to be completed within 2 hours
- ranked
  - top 10% of participants who played more than 1 game
  - 2nd of 19 in the league that played 5 or more games
  - 1st in an 8-game tournament involving 21 participants

Table of Contents

1 Background

2 Overview of Cicero

3 Method

4 Discussion and Direction for Future Research

5 Q and A

Cooperating and negotiating with humans on a complex task

- Cicero successfully combined strategic reasoning and dialogue to cooperate and negotiate with humans on a complex task
  - achieved strong human-level performance
  - passed as a human player for 40 games with 82 unique players
    - ⋆ no in-game message indicated players believed they were playing with an AI
    - ⋆ one player mentioned a suspicion post-game but did not lead to Cicero detected as an AI by other players in that league

Limitations and future research

- Filters reduced errors but Cicero occasionally sent messages that
  - contained grounding errors
  - contradicted its plans
  - or were otherwise strategically subpar
- these mistakes did not raise further suspicions that Cicero was an AI due to possibly
  - time pressure
  - humans occasionally make similar mistakes
- future work could explore Diplomacy with longer negotiation rounds
- strategically, Cicero reasoned purely in terms of actions for the current turn
- richer affordances of dialogue was limited due to intent representation, e.g.
  - strategic information revelation
  - asking questions
  - providing explanations
- *Diplomacy* provides a great platform to explore
  - connections between strategy and communication
  - with the goal of improving coordination between humans and agents

Implications for the future of ML

- It is striking the way in which Cicero achieved the deepst and most extensive integration of language and action[1]
  - unlike the popular view of 'end-to-end' ML
    - ⋆ a single general learning algorithm applies across the board
    - ⋆ with little internal structure and zero innate knowledge
  - deep learning systems tend to be less structured and less customised to particular problems
  - Cicero consists of hand-crafted modules with complex interactions
    - ⋆ a wide range of training material, including some built just for Cicero, some synethesised in programs hand-crafted by experts, are drawn upon
    - ⋆ uses a neurosymbolic approach in some aspects, e.g.
    - ⋆ association of messages in language with symbolic representation of actions,
    - ⋆ innate understanding of dialogue structure, etc
- "*If Cicero is any guide, machine learning may ultimately prove to be even more valuable if it is embedded in highly structured systems, with a fair amount of innate, sometimes neurosymbolic machinery.*"

[1] link to the article by Gary Marcus and Ernest Davis

Bonus Slide of Cicerro Explaining its Absence

Figure: Cicerro(France, Blue) explaining its absence due to technical difficulties.

Table of Contents

1. Background
2. Overview of Cicero
3. Method
4. Discussion and Direction for Future Research
5. Q and A

Q and A

Questions?