

## サッカー実況システム MIKE における実時間文章生成

田中久美子

橋田浩一

野田五十樹

電子技術総合研究所

## 概要

MIKEはRobocupサッカーシミュレーションシステムと連動し、日本語、英語、仏語の実況を実時間で生成するシステムである。本システムでは、選手やボールの位置などの原始的な入力情報を複数のプロセスが並行に解析し、結果を別のプロセスが整理し、内容を選択し、発話を行う。

実況における最大の課題は時機を逃さず発話をするという実時間性にあり、これは生成の研究ではこれまでにあまり議論されて来てはいない。MIKEでは実時間での発話を制御するために、発話内容の候補に重要度を割り振り、実況を通して得られる全重要度を最大にするという原則のもとに内容選択を行う。この原則下では割り込みや省略といった表層生成が、話題選択と不可分に実装される。

出力実況の例を示すと共に、簡単な評価を示す。

## 1 はじめに

人間の発話の重要性を決める要因には、その内容が状況に応じて適切である事だけでなく、適切な時機に発話される事も含まれる。ところが、これまでの発話の研究では、適切な内容を生成することに研究の焦点が当てられがちであり、実時間的側面はあまり考慮されてこなかった。その理由の一つは、これまでの発話研究の応用システムは人間と機械の対話を対象としたものが多く、ここでは人間がシステムの応答を待つことができたことが挙げられるであろう。

一方で自動実況という研究課題には、試合進行という強力な時間的制約がかかっている。発話すべき時に発話しなければ、試合は展開し、異なった状況へ進んで行ってしまう。シュートが起きた時にシュートと言わなければ、シュートに関する実況の重要性が著しく下がる。すなわち、実況は実時間性を議論するための適切な応用問題の一つなのである。

過去の自動実況に類する研究としては、画像を自然言語で描写することを試みたVitraシステムがある[5][1][2]。その一貫として、サッカーを動画画像から解析した上で自然言語生成で状況を描写することを試みている。本研究はこのシステムと比べて、より実時間性の研究に焦点を当てており、また対象物の描写に留まらず、いろいろな文脈からの説明を試みるという2点において異なっている。

MIKEはRobocupサッカーシミュレーションシステムに対して英語、日本語、フランス語で実況を自動生成するシステムである。サッカーをその題材として選んだ理由はつぎのとおり。第一にサッカーでは複数の出来事が並行して起こっており、発話内容を取捨選択するという観点からは適切な題材である点である。第二にRobocup Soccerの研究により、試合の時々刻々の選手とボールの位置とい

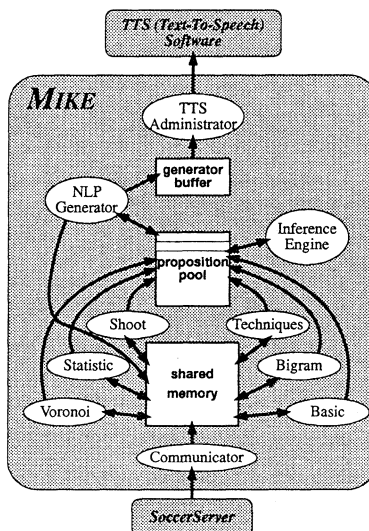


図 1: MIKE の設計

う高品質な入力データが存在することである。

## 2 MIKE の設計

MIKE('Multi-agent Interactions Knowledgeably Explained'の略)は、サッカーサーバー<sup>1</sup>と連動して実時間で実況を行う。サッカーサーバーは100msecに一度、試合のログ情報をクライアントに送信する。ログ情報には、

- 全選手及びボールの位置
- 得点およびプレーモード(ゴールキック、プレーオンなど)

が含まれ、システムはこの原始的な情報から試合状況を認識、推論、取捨選択して解説を行う。

MIKEの内部実装は図1に示すように役割分担型である。図の楕円は並行動作するプロセスを、四角はデータ格納領域を表している。このような設計は自然言語分野では、黒板方式としてHearsay-IIなどで用いられてきた[3]。MIKEの設計もそれに準ずるものではあるが、さらにプロセス同士で独自の通信を行うことができる。

エージェント間通信は実況の断片を内部的に表現する

<sup>1</sup>サッカーサーバーとはマルチエージェントシステムの評価基準として野田ら[6]によって提案された。

<http://ci.etl.go.jp/moda/soccer/server.html> 参照。

表 1: Proposition の例

	local	global
event	Kick Pass Dribble ShootPredict	ChangeForm  SideChange
state	Mark PlayerPassSuccessRate ProblematicPlayer PlayerActive	TeamPassSuccessRate AveragePassDistance Score Time

Proposition (以下 Prop と記述) を介して行われる。一つの Prop はタグと属性から成り、表 1 に例を示した。たとえば、5 番選手のキックは (Kick 5) と表現され、Kick はタグ、5 は属性となる。現状では MIKE は約 60 の Prop を使用している。

MIKE 中の各プロセスの動きをつぎに列挙する。

- サッカー解析エージェントには 6 つあり、並行にサッカーを異なる方式で認識し、結果に応じた Prop を生成し、図中の Pool というデータ領域にそれを格納する。内 3 つは図中の 'Basic,' 'Technic,' 'Shoot' であり、フィールド内の基本的なキックからパス、バックパスなどの高度技術、またシュートに関連する情報を認識する。これに対し、残りの 3 つは図中の 'Bigram,' 'Voronoi,' 'Statistic' であり、キック系列の遷移に基づく選手の活躍状況の解析、ボロノイ図を用いた選手の守備範囲の解析および配置の評価、および統計を用いた陣形の解析など、より包括的な視点からの解析となる (詳細は [7] を参照)。
- 推論エンジンは解析エージェントの結果の推論を行う。ここでより高度な帰結、Prop 間の関係付けが行われる (§3.2 章参照)。
- 自然言語エージェントは状況 (サッカーの状況、および実況の状況) に応じて Pool より一つ Prop を選ぶ。これを各 Prop に対して複数用意されているテンプレートに対応させ、自然言語変換する。

現状では音声合成ソフトウェアは日本語に関してのみ MIKE に付加されている [4]。

MIKE の発話内容は、6 つの解析エージェントの解析結果を統合することにより得られる。図 2 に現状の発話可能な内容を列挙する。

### 3 推論, 選択, 及び生成

#### 3.1 Proposition の重要度

Prop には、状況に応じてどの程度重要かを示す得点を付加しており、重要度と呼んでいる。Prop の重要度は図 3 に示すように変化する。Pool に入れた後は重要度は時間と共に減衰し、重要度が 0 になった時に Pool より削除される。減衰に関しては、局所的な内容を持つ Prop は急速に減衰するものとした。Prop が自然言語生成プロセスに選択されて言及された場合、あるいは推論に使用された場合には、一つの Prop に対する冗長な処理を省くために重要度は一定割合で減少させる。

- チームの背景
- チームの現状 (観衆がわからないことを解説)  
陣形の変化、ポジション替え、高度な技の指摘
- チームの力量の評価  
問題選手の指摘、優良選手の指摘、平均陣形の評価  
ある瞬間の陣形の評価、必勝パターンの指摘、  
無駄な動き指摘、特定な選手の配置の評価
- チームを強くするための改良法の提案  
手薄な場所の指摘、問題児の配置替え
- ゲーム展開の予測  
必勝パターンに基づく得点の予測、  
ボールを取りに行くべき選手の予測、  
パスの方向の予測
- 明解な内容の実況

図 2: Mike の発話内容

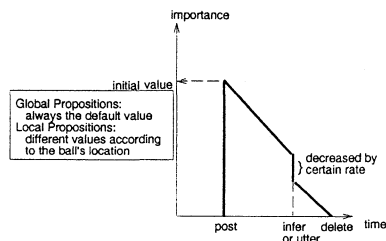


図 3: Proposition の重要度の変化

重要度はサッカー解析エージェントにおいて初期化される。ゴール周囲ではパスワークに焦点を絞った実況をし、フィールド中央ではより大局的な実況をするという文脈制御のために重要度の初期化にはつぎのような工夫をほどこしている。局所的な内容の Prop はボールの位置に応じて初期化が異なり、ゴールに近いほど高く、フィールドの中央に位置するほど低く初期化する。大局的な Prop は常に一定値に初期化される。

更に、実況がフィールドの状況のみならず、実況の状況をも反映するために MIKE の現在の実況内容を解析エージェントに知らせる。解析エージェントはその実況内容を含むような Prop を生成する際には通常的重要度の初期値よりも高い値で初期化する。たとえば、実況内容が 5 番選手に関するものである場合には、解析エージェントは 5 番選手を含む Prop の重要度を通常値よりも高い値で初期化する。

#### 3.2 推論

Pool に登録されたばかりの Prop は解析結果の集合に過ぎないが、これを整理するのが推論エンジンである。推論エンジンは Pool から重要度最大の Prop を一つ取り出し、推論規則を適用して Pool に戻すことを繰り返すだけのプロセスである。推論規則としては以下のものが用意されている。

変換: たとえば、

(PassSuccessRate 7 30) (PassPattern 7 goal)  
→ (Active 7)

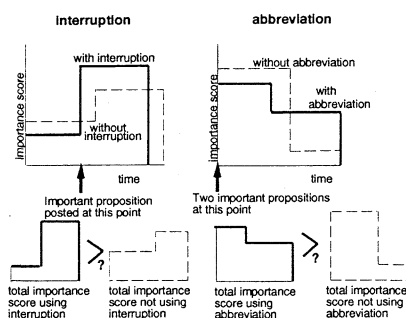


図 4: 割り込み, 省略処理の有無による重要度の変化

7番選手が活躍していると言えるのは、その選手のパス成功率が高く、さらに何回もゴールを決めている場合であることを意味している。

関係付け: 関係として、理由、結果、並列、逆説、が用意されている。たとえば  $p_1$  を (PassSuccessRate 7 30),  $p_2$  を (PlayerOnVoronoiLine 7) とすると、

$$p_1 p_2 \rightarrow (\text{Reason } p_1 p_2)$$

と関係付けする。7番選手が高いパス成功率を示すのは、7番選手が良い配置、すなわち、敵のチームの Voronoi 点上に配置されていることを意味する ([7] 参照)。

削除:  $p_1 p_2 \rightarrow p_1$  という処理を指す。たとえば、

$$(\text{Pass } 3 \ 4)(\text{Kick } 3) \rightarrow (\text{Pass } 3 \ 4)$$

は、三番から4番へのパスが言及された時には三番のキックに関しては言及する必要がないことを意味する。削除は左辺のある Prop が別の Prop に内容的に包括されるか、あるいは矛盾する場合に適用される。推論後の Prop の重要度は現状では左辺の Prop の重要度の線形結合の値として計算している。

### 3.3 話題選択と言語生成

重要度は、その時点時点において、観客にとってより興味深いと想定される Prop はほど大きい値を持つように設計している。したがって、Prop を Pool から一つ選ぶ戦略として、実況によって得られる全重要度の最大化を原則としている。この原則からは、選ぶ時点で重要度が最大の Prop を選んで発話するのが基本的な処理となる。

ところが、選択時に重要度最大の Prop を選んでいるだけでは全重要度最大化には不十分である。その原因は、選んで生成してから、音声合成器に送って言い終わるまでの時間差である。音声合成器が発話をしている間に試合は展開し、複数の Prop が Pool に挿入されてくる。

図4は、立て軸が単位時間あたりの重用度、横軸が時間を表わし、実線や点線と横軸とで囲む面積が得られる全重用度となる。今、発話途中に、重要な Prop が Pool に登録されたとする(図左)。第一の方策としては、発話が終わのを待ってから新しい Prop にとりかかる方法である(点線)。Prop は時間と共に重用度が減衰するので、新しい Prop を発話する時には、時間差分得られる重用度が減っている。第二の方策としては、発話を中断してすぐに新しい Prop の発話にとりかかる割り込み法である(実線)。この場合は、前の話は途中で終わってしまうので、それまで

に得られる重要度自体が減るが、その分つぎの Prop からは最大の重用度を得られる。したがって、割り込みを行うか否かは点線および実線と横軸で囲まれる二つの面積を比較し、大きい方を選択することによって決定される。

省略も同様に考えることができる。今、同じ程度の重要度を持つ Prop がほぼ同時に Pool に入っているとすると(図右)。まず一つ選んで発話し、そのつぎに二つ目にとりかかるが、一つ目にどの程度時間がかかるかによって、二つ目から得られる重要度は変化する。一つ目を省略しない場合は、二つ目は時間がたってから発話されるので、重要度は随分と減っている(点線)。一方、一つ目を省略する場合は、二つ目により早くとりかかるので、重要度の減りは押えられる。しかし、この場合は一つ目を省略した表現にするために、ひとつ目の Prop から得られる重要度は通常より減るであろう(実線)。省略を行うか否かは点線及び実線と、横軸とで囲まれる面積を比較することによって行われる。繰り返しもまったく同様に考えることができる。

このように、単に選択の時点で重要な Prop を選択しているだけでは、原則の「全重要度の最大化」を満たさず、補正として割り込みや省略といった表層処理を行う必要がある。

### 3.4 How, What, When-to-Say

自然言語の発話の分野では How-to-say 処理と What-to-say 処理は別の処理であるか否かということが議論されてきた [8]。一つ論点としては What と How は全く同一のプロセスとして実現できるのか、あるいは互いに通信し合う別のプロセスとして実現すべきであるのかという点がある。

MIKE では、What 処理とはサッカーの解析エージェント、推論、自然言語エージェントによる Prop の選択部分であり、How 処理は割り込み、省略、繰り返し処理決定、テンプレート選択による自然言語生成部分となる。MIKE では What と How は不可分なプロセスとして実装されている部分と、相互通信する別のプロセスとして実装されている部分が混在する。不可分となっているのは、話題の選択処理と割り込み、省略処理である。時機を逸せずに発話を行うには、Prop を選ぶと同時に割り込み処理を行うか否かが決まり、別のプロセスとする利点は全く無い。

When-to-say という制約がなければ、MIKE ではそもそも割り込み処理などは考えにくく、How と What は互いに通信し合う別のもので十分である。つまり、When-to-say 制約こそが、How と What 処理を不可分なモジュールとする一つの根拠となっていると考えられる。

## 4 評価

### 4.1 出力例

日英のみ、Robocup'97 の準々決勝の4点目のゴールの付近の出力例を示す(図5, 図6)。実況はサッカーシミュレーション画面を見ながら聞くことが想定されており、試合の展開を逐一説明してはいない。したがって、必要に応じて括弧中に試合の説明を付加した。尚、MIKE は並行処理を行っているため、出力はそのたびごとに異なる。

### 4.2 割り込み、省略処理と全重要度の増分

準々決勝のある時間帯において得られた重用度を図7に示す。実線が割り込み、省略、繰り返しを用いた時、点

Red3 collects the ball from Red4, Red3, Red-Team, wonderful goal! 2 to 2! Red3's great center shot! Equal! The Red-Team's formation is now breaking through enemy line from center, The Red-Team's counter attack, Red3's goal! Kick off, Yellow-Team, Red1 is very active because, Red1 always takes good positions, Second half of Robocup-97 quarterfinal(予め入力されている背景データが出力されている). Left is Ohta Team, Japan, Right is Humboldt, Germany, Red1 takes the ball, bad pass, (黄色チームは赤チームにキックオフの直後にボールを奪われた) Interception by the Yellow-Team, Wonderful dribble, Yellow2, Yellow2 (黄色6が黄色2をガードしている), Yellow6's pass, A pass through the opponents' defense, Red6 can take the ball,because, Yellow6 is being marked by Red6, .... The Red-Team's counter attack, The Red-Team's formation is (割り込み), Yellow5, Back pass of Yellow10, Wonderful pass,

図 5: MIKE RoboCup'97 準々決勝に対する出力例

赤 4 番のプレー, 赤 4 番から赤 3 番, 赤 3 番, 赤チームの得点です, 2 点です, 2 対 2, 赤 3 番のセンターからのシュート! 同点! 赤 3 番がゴール, 赤チームは中央突破型です, 黄色チームの速攻プレー, ロボカップ'97 準々決勝後半戦です。黄色は太田チーム, 日本, 赤はフンボルト, ドイツ 赤 1 番が取れそう, 黄色チーム (黄色が赤チームよりボールを取り上げ, 黄色 2 番と黄色 6 番がボールを持っている), 黄色 5 番の方向, 赤 6 番が取れそう, なぜなら, 赤 6 番は黄色 6 番をマークしています, 赤 5 番が取れそう (予測が行われたが, ボールはフィールドから出てしまった), 赤チームのキックイン 赤 8 番, 赤 8 番, 黄色 8 番です。今度もまたシュートにつなげられるでしょうか (黄色 8 番からパスを重ねてシュートに至るパターンがすでに抽出されており, それに基づいて予測を行っている)

図 6: 日本語の出力例

線が用いない時である。各ピークはその時に Prop を選択されたことを示す。

- 実線よりも点線の方がピークが遅れがちである。
- 実線で囲む面積が点線で囲む面積よりも大きい。

の二点が観察される。面積の増分を全体にこの試合全体に渡って計算したところ、実線の方が 9.9%<sup>2</sup>増大した。

## 5 結論

サッカーの試合の実況を生成する MIKE システムを紹介し、文章生成部分に関して主に論じた。実況における研究の観点は時機を逃さず発話するという実時間性にある。

MIKE では選手やボールの位置情報といった原始的な入力に対しさまざまな観点から複数の解析エージェントが解析を行う。この結果を推論エージェントが整理整頓し、自然言語生成エージェントが状況に最も即した結果を選んで発話に至る。この時、状況に合致する内容であるか否かを測る尺度として発話内容には重要度を付加している。これに基づき、話題の選択をする戦略として実況を通して最終的に得られた全重要度を最大化するという原則を用いている。この原則下では、発話内容を決定する話題の選択と割り込み、省略といった表層的な文章生成が、自然に不可分なモジュールとして実装される。最後に MIKE

<sup>2</sup>増える度合はシステム中のパラメータに依存する。

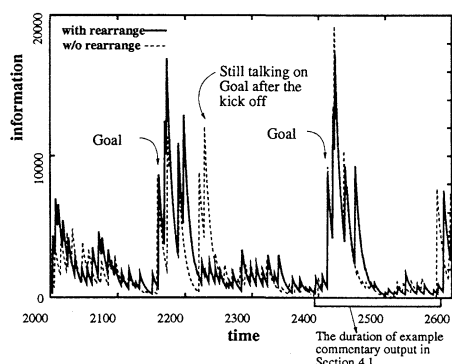


図 7: 割り込み等の有無による重要度の変化 (準々決勝)

の出力結果と簡単な評価を示した。

課題の第一はシステム内のパラメータの学習である。理想的な実況コーパスが存在せず、入力コーパス自体も多くは存在しない現状では、出力を評価するのが難しい。そこで、MIKE の出力に対して人為的な評価を行い、それを自動的にシステム内のパラメータに反映させる方法を考案中である。

第二に時間表現、指示表現、節や句の埋め込みといった、より高度な自然言語生成モジュールを作る必要があるであろう。

最後に Vitra システムのように、実世界の実況に役立てる方向の研究があろう。本システムの入力としては、原始的な位置情報で十分なので、この観点からはトラッキングビジョン [9] などとの連携により実世界サッカーの中継に役立てることも将来的に考えられるであろう。

## 参考文献

- [1] E. André, G. Herzog, and T. Rist. On the simultaneous interpretation of real world image sequences and their natural language description: The system soccer. In *Proc. of the 8th ECAI*, pages 449–454, Munich, 1988.
- [2] E. André, G. Herzog, and T. Rist. Multimedia presentation of interpreted visual data. In P. McKevitt, editor, *Proceedings of AAAI-94, Workshop on Integration of Natural Language and Vision Processing*, pages 74–82, Seattle, WA, 1994.
- [3] L. D. Erman, F. Hayes-Roth, V. R. Lesser, and D. R. Reddy. The Hearsay-II speech understanding system: Integrating knowledge to resolve uncertainty. *ACM Computing Surveys*, 12(2):213–253, 1980.
- [4] Fujitsu. *FSUNvoice1.0 Japanese speech synthesizer Document*, 1995.
- [5] G. Herzog and P. Wazinski. Visual TRANslator: Linking perceptions and natural language descriptions. In *Artificial Intelligence Review*, volume 8, pages 175–187, 1994.
- [6] I. Noda and H. Matubara. Soccer Server and researches on multi-agent systems. In Hiroaki Kitano, editor, *Proceedings of IROS-96 Workshop on RoboCup*, pages 1–7, Nov. 1996.
- [7] K. Tanaka-Ishii, I. Noda, I. Frank, H. Nakashima, K. Hasida, and H. Matsubara. Mike: An automatic commentary system for soccer. In *Proceedings of ICMA98*, Paris, France, 1998.
- [8] 乾健太郎 徳永建伸. 1980 年代の自然言語生成 -1,2,3-. 人工知能学会誌, 6(3–5), 1991.
- [9] 森田俊彦他 内山隆. 動き追跡処理システム「トラッキングビジョン」. 日本ロボット学会誌, 16(1):52–53, 1997.